

銘傳大學

資訊管理學系碩士班 碩士論文

應用蟻群演算法建構 LSB 替代矩陣之研究



研究生：劉妍芝

指導教授：許慶昇 博士

中華民國九十八年六月

應用蟻群演算法建構 LSB 替代矩陣之研究

研究生：劉妍芝

指導教授：許慶昇 博士

銘傳大學資訊管理學系碩士班

摘 要

最低位元置換法(Least Significant Bit Substitution, LSB Substitution)，是一種資訊隱藏(Information Hiding)的方法，它能夠將秘密訊息藏入一張掩護影像(Cover Image)之中，以防止被惡意人士發現並竊取秘密訊息。然而，由於最低位元置換法的秘密訊息隱藏方式過於簡單，且藏入秘密訊息後會破壞偽裝影像，因此，有學者提出替代矩陣來改善偽裝影像的影像品質，並增加秘密訊息的安全性。本質上，求解替代矩陣是一個組合最佳化的問題。目前，已有許多最佳化演算法被應用來建構替代矩陣，例如貪婪演算法、基因演算法等。而蟻群最佳化演算法(Ant Colony Optimization Algorithm, ACO Algorithm)也是一種最佳化方法，它是由自然界螞蟻的行為所得到的啟發。經研究發現，螞蟻之間藉由一種名為「費洛蒙(Pheromones)」的化學物質做為溝通媒介，能夠幫助蟻群在最短時間內找到巢穴與食物地點之間的最短路徑。目前，蟻群最佳化演算法在一些時間複雜度高的組合問題上，已經有很好的成效，且其效能優於基因演算法(Genetic Algorithm)。因此，基於資訊隱藏的執行效率考量，本研究將結合蟻群最佳化演算法來建構 LSB 的替代矩陣，以降低求解替代矩陣的時間複雜度。此外，我們也提出了位置替代矩陣來進一步改善偽裝影像的影像品質；因此，除了單獨使用傳統的顏色替代矩陣之外，我們的方法同時結合顏色與位置替代矩陣，來將秘密訊息隱藏於掩護影像之中。實驗結果顯示，加入位置替代矩陣能有效的改善偽裝影像的影像品質。

關鍵詞：資訊隱藏、蟻群最佳化演算法、最低位元置換法、替代矩陣。

Study of the Construction of LSB Substitution

Matrices Based on Ant Colony Optimization Algorithms

Student: Yen-Chih Liu Advisor: Dr. Ching-Sheng Hsu
Department of Information Management
Ming Chuan University

Abstract

Least Significant Bit (LSB), a kind of information hiding techniques, can embed a secret into a cover image to prevent the revelation of secrets to malicious people. Since the method is quite simple and it will destroy cover images, the substitution matrix is proposed to improve the image quality of cover images and the security of secrets. Basically, the construction of optimal substitution matrices is intrinsically a combinatorial optimization problem. Nowadays, some optimization methods have been applied to the construction of substitution matrices, such as greedy algorithms and genetic algorithms. Ant Colony Optimization (ACO) Algorithm, which is also a kind of optimization methods, was derived from the observation of real ants' behavior. Researchers found that real ants use the chemicals called "pheromones" to communicate with each other and form the shortest path connecting a nest of ants and a food source. It has been shown that ACO Algorithm is effective to solve many combinatorial optimization problems and its efficiency is better than that of genetic algorithm and simulated annealing. Therefore, in this thesis, the ACO algorithm will be exploited to construct optimal substitution matrices for efficiency consideration. In addition, we also propose position substitution matrices to further improve the image quality. Thus, instead of using only the color substitution matrix, we combine color and position substitution matrices to conceal secrets in cover images. Experimental results show that the addition of position matrices is effective in improving the image quality of cover images.

Keywords : Information Hiding, Ant Colony Optimization Algorithm, Least Significant Bit, Substitution Matrix.



誌 謝

光陰似箭，歲月如梭，回想起剛進入研究所的日子，真是既擔心又高興，擔心在於能否應付繁雜的課業及研究，而高興在於終於成為研究生，能繼續朝著夢想前進。如今，研究所課業已修習完畢，而最重要的論文也終於完稿了，真是令我高興不已，回想起這兩年的日子，要是沒有指導老師許慶昇的諄諄教誨，及超越常人的耐心教導，並且不時的關心及安慰，恐怕是無法將論文完稿，所以真的真的非常感謝您。

接著，我要感謝的是同門的同學尹良，因為在這兩年的日子中，與我共同學習，遇到困難時，總能在旁協助我解決問題；還有要感謝學長奕煌及學弟顯仁，在研一的日子中，學長給予我許多的經驗與教導，而在研二的日子中，顯仁總是在旁協助處理事務，使我能省去需多的時間，當然還有 204 研究室所有的同學及學弟妹，謝謝大家的幫助，使我在課業上能夠順利的修習完畢。

目錄

圖目錄	vii
表目錄	ix
第壹章 緒論	1
1.1 研究背景	1
1.2 研究動機	4
1.3 研究目的	6
1.4 研究架構	6
第貳章 文獻探討	7
2.1 最低位元置換法 (Least Significant Bit Substitution)	7
2.1.1 簡單最低位元置換法 (Simple-LSB Substitution)	7
2.1.2 最佳最低位元置換法	8
2.1.3 利用基因演算法的最低位元置換法	9
2.1.4 基於DES和貪婪演算法的最低位元置換法	13
2.1.5 基於JPEG和粒子群演算法的最低位元置換法	15
2.2 蟻群最佳化(Ant Colony Optimization)方法之演進	20
2.2.1 自然界螞蟻	20
2.2.2 人工螞蟻	21
2.2.3 蟻群最佳化次經驗(ACO-Metaheuristic)	23
2.2.4 蟻群最佳化演算法	25
2.2.5 蟻群最佳化演算法目前應用的範圍	27
第參章 研究方法	28
3.1 問題描述	28
3.2 符號定義	29
3.3 秘密訊息的隱藏	31
3.4 最低位元置換法	33
3.5 替代矩陣	34
3.6 結合像素與位置替代矩陣及其討論	36
3.7 運用蟻群演算法找尋替代矩陣	38
3.7.1 蟻群最佳化演算法與替代矩陣之間的關聯	38
3.7.2 以蟻群演算法求解替代矩陣	39
3.8 秘密訊息的取出	42
第肆章 實驗結果與分析	44
4.1 實驗簡介	44
4.2 掩護影像的每個像素藏入 2 位元秘密訊息的實驗結果	44
4.3 掩護影像的每個像素藏入 4 位元秘密訊息的實驗結果	62
4.4 實驗討論	87

第伍章	結論.....	94
參考文獻	96
著作	99



圖目錄

圖 1 資訊隱藏分類圖	2
圖 2 最佳最低位元置換法的流程	8
圖 3 基因演算法的最低位元置換法流程	10
圖 4 基因演算法與替代矩陣關係圖解	11
圖 5 基因演算法交配過程範例	12
圖 6 基於DES和貪婪演算法的最低位元置換法流程圖	13
圖 7 DES加密步驟	14
圖 8 JPEG標準量化表	16
圖 9 藏入秘密訊息的順序	16
圖 10 JQTM方法的量化表	17
圖 11 基於JPEG和粒子群演算法的最低位元置換法嵌入秘密訊息流程	18
圖 12 基於JPEG和粒子群演算法的最低位元置換法所提議的量化表	18
圖 13 巢穴到食物區的路徑圖	20
圖 14 推測演算法的圖解	23
圖 15 蟻群最佳化次經驗的虛擬程式碼	24
圖 16 秘密訊息隱藏流程	31
圖 17 秘密訊息轉換範例	32
圖 18 最低位元置換法範例	33
圖 19 (A) 不合格的替代矩陣；(B) 合格的替代矩陣	35
圖 20 顏色替代矩陣轉換範例	35
圖 21 位置替代矩陣轉換範例	36
圖 22 顏色替代矩陣與位置替代矩陣關係之範例	37
圖 23 4×4 替代矩陣的蟻群最佳化演算法問題圖	39
圖 24 秘密訊息偵測流程	42
圖 25 AIRPLANE蟻蟻迭代次數與PSNR值分佈圖（藏入 2 位元）	45
圖 26 BABOON蟻蟻迭代次數與PSNR值分佈圖（藏入 2 位元）	46
圖 27 BIRD蟻蟻迭代次數與PSNR值分佈圖（藏入 2 位元）	47
圖 28 BIRD1 蟻蟻迭代次數與PSNR值分佈圖（藏入 2 位元）	48
圖 29 BOAT蟻蟻迭代次數與PSNR值分佈圖（藏入 2 位元）	49
圖 30 CAT蟻蟻迭代次數與PSNR值分佈圖（藏入 2 位元）	50
圖 31 GIRL蟻蟻迭代次數與PSNR值分佈圖（藏入 2 位元）	51
圖 32 LENA蟻蟻迭代次數與PSNR值分佈圖（藏入 2 位元）	52
圖 33 LENNA蟻蟻迭代次數與PSNR值分佈圖（藏入 2 位元）	53
圖 34 MANDARIN蟻蟻迭代次數與PSNR值分佈圖（藏入 2 位元）	54
圖 35 PEPPER蟻蟻迭代次數與PSNR值分佈圖（藏入 2 位元）	55
圖 36 TIFFANY蟻蟻迭代次數與PSNR值分佈圖（藏入 2 位元）	56

圖 37 TIGER 螞蟻迭代次數與PSNR值分佈圖 (藏入 2 位元)	57
圖 38 TOYS 螞蟻迭代次數與PSNR值分佈圖 (藏入 2 位元)	58
圖 39 ZELDA 螞蟻迭代次數與PSNR值分佈圖 (藏入 2 位元)	59
圖 40 銘傳大學螞蟻迭代次數與PSNR值分佈圖 (藏入 2 位元)	60
圖 41 銘傳資管螞蟻迭代次數與PSNR值分佈圖 (藏入 2 位元)	61
圖 42 AIRPLANE 螞蟻迭代次數與PSNR值分佈圖 (藏入 4 位元)	63
圖 43 BABOON 螞蟻迭代次數與PSNR值分佈圖 (藏入 4 位元)	64
圖 44 BIRD 螞蟻迭代次數與PSNR值分佈圖 (藏入 4 位元)	65
圖 45 BIRD1 螞蟻迭代次數與PSNR值分佈圖 (藏入 4 位元)	66
圖 46 BOAT 螞蟻迭代次數與PSNR值分佈圖 (藏入 4 位元)	67
圖 47 CAT 螞蟻迭代次數與PSNR值分佈圖 (藏入 4 位元)	68
圖 48 GIRL 螞蟻迭代次數與PSNR值分佈圖 (藏入 4 位元)	69
圖 49 GOLD 螞蟻迭代次數與PSNR值分佈圖 (藏入 4 位元)	70
圖 50 LENA 螞蟻迭代次數與PSNR值分佈圖 (藏入 4 位元)	71
圖 51 LENNA 螞蟻迭代次數與PSNR值分佈圖 (藏入 4 位元)	72
圖 52 MONALISA 螞蟻迭代次數與PSNR值分佈圖 (藏入 4 位元)	73
圖 53 SAILBOAT 螞蟻迭代次數與PSNR值分佈圖 (藏入 4 位元)	74
圖 54 TIFFANY 螞蟻迭代次數與PSNR值分佈圖 (藏入 4 位元)	75
圖 55 TOYS 螞蟻迭代次數與PSNR值分佈圖 (藏入 4 位元)	76
圖 56 ZELDA 螞蟻迭代次數與PSNR值分佈圖 (藏入 4 位元)	77
圖 57 銘傳大學螞蟻迭代次數與PSNR值分佈圖 (藏入 4 位元)	78
圖 58 銘傳資管螞蟻迭代次數與PSNR值分佈圖 (藏入 4 位元)	79
圖 59 AIRPLANE 螞蟻迭代次數與PSNR值分佈圖 (藏 4 位元; 秘密訊息 2)	80
圖 60 BABOON 螞蟻迭代次數與PSNR值分佈圖 (藏 4 位元; 秘密訊息 2)	81
圖 61 BIRD1 螞蟻迭代次數與PSNR值分佈圖 (藏入 4 位元; 秘密訊息 2)	82
圖 62 BOAT 螞蟻迭代次數與PSNR值分佈圖 (藏入 4 位元; 秘密訊息 2)	83
圖 63 GIRL 螞蟻迭代次數與PSNR值分佈圖 (藏入 4 位元; 秘密訊息 2)	84
圖 64 LENNA 螞蟻迭代次數與PSNR值分佈圖 (藏入 4 位元; 秘密訊息 2)	85
圖 65 PEPPER 螞蟻迭代次數與PSNR值分佈圖 (藏入 4 位元; 秘密訊息 2)	86
圖 66 4.2 節的實驗數據的平均值	88
圖 67 4.3 節藏入第一種秘密訊息的實驗數據的平均值	91
圖 68 4.3 節藏入第二種秘密訊息的實驗數據的平均值	92

表目錄

表 1 AIRPLANE影像介紹 (藏 2 位元)	45
表 2 AIRPLANE實驗數據 (藏 2 位元)	45
表 3 BABOON影像介紹 (藏 2 位元)	46
表 4 BABOON實驗數據 (藏 2 位元)	46
表 5 BIRD影像介紹 (藏 2 位元)	47
表 6 BIRD實驗數據 (藏 2 位元)	47
表 7 BIRD1 影像介紹 (藏 2 位元)	48
表 8 BIRD1 實驗數據 (藏 2 位元)	48
表 9 BOAT影像介紹 (藏 2 位元)	49
表 10 BOAT實驗數據 (藏 2 位元)	49
表 11 CAT影像介紹 (藏 2 位元)	50
表 12 CAT實驗數據 (藏 2 位元)	50
表 13 GIRL影像介紹 (藏 2 位元)	51
表 14 GIRL實驗數據 (藏 2 位元)	51
表 15 LENA影像介紹 (藏 2 位元)	52
表 16 LENA實驗數據 (藏 2 位元)	52
表 17 LENNA影像介紹 (藏 2 位元)	53
表 18 LENNA實驗數據 (藏 2 位元)	53
表 19 MANDARIN影像介紹 (藏 2 位元)	54
表 20 MANDARIN實驗數據 (藏 2 位元)	54
表 21 PEPPER影像介紹 (藏 2 位元)	55
表 22 PEPPER實驗數據 (藏 2 位元)	55
表 23 TIFFANY影像介紹 (藏 2 位元)	56
表 24 TIFFANY實驗數據 (藏 2 位元)	56
表 25 TIGER影像介紹 (藏 2 位元)	57
表 26 TIGER實驗數據 (藏 2 位元)	57
表 27 TOYS影像介紹 (藏 2 位元)	58
表 28 TOYS實驗數據 (藏 2 位元)	58
表 29 ZELDA影像介紹 (藏 2 位元)	59
表 30 ZELDA實驗數據 (藏 2 位元)	59
表 31 銘傳大學影像介紹 (藏 2 位元)	60
表 32 銘傳大學實驗數據 (藏 2 位元)	60
表 33 銘傳資管影像介紹 (藏 2 位元)	61
表 34 銘傳資管實驗數據 (藏 2 位元)	61
表 35 AIRPLANE影像介紹 (藏 4 位元)	63
表 36 AIRPLANE實驗數據 (藏 4 位元)	63

表 37 BABOON影像介紹 (藏 4 位元)	64
表 38 BABOON實驗數據 (藏 4 位元)	64
表 39 BIRD影像介紹 (藏 4 位元)	65
表 40 BIRD實驗數據 (藏 4 位元)	65
表 41 BIRD1 影像介紹 (藏 4 位元)	66
表 42 BIRD1 實驗數據 (藏 4 位元)	66
表 43 BOAT影像介紹 (藏 4 位元)	67
表 44 BOAT實驗數據 (藏 4 位元)	67
表 45 CAT影像介紹 (藏 4 位元)	68
表 46 CAT實驗數據 (藏 4 位元)	68
表 47 GIRL影像介紹 (藏 4 位元)	69
表 48 GIRL實驗數據 (藏 4 位元)	69
表 49 GOLD影像介紹 (藏 4 位元)	70
表 50 GOLD實驗數據 (藏 4 位元)	70
表 51 LENA影像介紹 (藏 4 位元)	71
表 52 LENA實驗數據 (藏 4 位元)	71
表 53 LENNA影像介紹 (藏 4 位元)	72
表 54 LENNA實驗數據 (藏 4 位元)	72
表 55 MONALISA影像介紹 (藏 4 位元)	73
表 56 MONALISA實驗數據 (藏 4 位元)	73
表 57 SAILBOAT影像介紹 (藏 4 位元)	74
表 58 SAILBOAT實驗數據 (藏 4 位元)	74
表 59 TIFFANY影像介紹 (藏 4 位元)	75
表 60 TIFFANY實驗數據 (藏 4 位元)	75
表 61 TOYS影像介紹 (藏 4 位元)	76
表 62 TOYS實驗數據 (藏 4 位元)	76
表 63 ZELDA影像介紹 (藏 4 位元)	77
表 64 ZELDA實驗數據 (藏 4 位元)	77
表 65 銘傳大學影像介紹 (藏 4 位元)	78
表 66 銘傳大學實驗數據 (藏 4 位元)	78
表 67 銘傳資管影像介紹 (藏 4 位元)	79
表 68 銘傳資管實驗數據 (藏 4 位元)	79
表 69 AIRPLANE影像介紹 (藏 4 位元; 秘密訊息 2)	80
表 70 AIRPLANE實驗數據 (藏 4 位元; 秘密訊息 2)	80
表 71 BABOON影像介紹 (藏 4 位元; 秘密訊息 2)	81
表 72 BABOON實驗數據 (藏 4 位元; 秘密訊息 2)	81
表 73 BIRD1 影像介紹 (藏 4 位元; 秘密訊息 2)	82
表 74 BIRD1 實驗數據 (藏 4 位元; 秘密訊息 2)	82

表 75 BOAT影像介紹（藏 4 位元；秘密訊息 2）	83
表 76 BOAT實驗數據（藏 4 位元；秘密訊息 2）	83
表 77 GIRL影像介紹（藏 4 位元；秘密訊息 2）	84
表 78 GIRL實驗數據（藏 4 位元；秘密訊息 2）	84
表 79 LENA影像介紹（藏 4 位元；秘密訊息 2）	85
表 80 LENA實驗數據（藏 4 位元；秘密訊息 2）	85
表 81 PEPPER影像介紹（藏 4 位元；秘密訊息 2）	86
表 82 PEPPER實驗數據（藏 4 位元；秘密訊息 2）	86



第壹章緒論

1.1 研究背景

隨著時代的進步，資訊技術也快速的蓬勃發展，其中，網際網路的迅速發展帶給資訊技術很大的衝擊與影響，雖然網際網路可以使人們變得省時省力，處理許多事務只需擁有電子產品和網路即可，但是相反的，網際網路卻使得一些機密資料變的更加危險，容易被惡意者藉由網際網路來竊取，或隨意流傳出去，因此，資訊隱藏變成很重要的關鍵。因為如果能夠將資訊隱藏起來，就不容易被有心人士發覺，可以增加機密資料的安全性(Aslantas & Ozer, 2008; Das & Maitra, 2006; Wang et al., 2007; Zhang & Cui, 2006)。

而資訊隱藏(Information Hiding)這個議題並非現在才有的觀念(王旭正、柯建瑩, 2007; 黃義美、黃仁俊, 2001; Petitcolas et al., 1999)，早在中古世紀就有類似的觀念了。有一位希臘歷史學家 Herodotus 曾經提到，在中古世紀時，一位希臘人命令他最信任的僕人刮除頭髮，並將秘密訊息以刺青的方式留在頭皮上，直到頭髮長出來後將秘密訊息蓋住，而達到資訊隱藏的目的。或是在二次世界大戰的時候，德國間諜使用隱形墨水來傳遞敏感的重要軍事機密。在中國也有許多資訊隱藏的例子，像是古代在戰爭的時候，會將機密訊息寫在絲綢上面，然後捲成球狀並裹上一層蠟之後，再命人吞下傳遞，以增加機密訊息的安全性，或是將機密訊息變成「嵌字詩」之後再傳遞，所謂的「嵌字詩」指的是：古人多會利用詩詞文章描述心境或傳遞訊息，嵌字詩並非詩詞文章中表面所指的內容或意義，而是詩詞文章中每一句的第一個字或最後一個字所湊成的句子。例如清朝時期有一位文采出眾的文官紀曉嵐，為了諷刺一位貪贓枉法的大官，因此寫了一幅對聯送給這位大官，其內容是「家居化日光天下，人在春風和氣中」，仔細注意每一句的第一個字，得到的是「家人」二字，而「家人」指的是僕人的意思，因為此位大官本是一位大戶人家的僕人，只因阿諛奉承才當上大官，但卻為官不清廉，令人唾棄，所以特意捉弄這位大官。

早期的資訊隱藏技術大部分都是運用自然界的特性將資訊隱藏起來，但現今由於資訊快速的蓬勃發展，資料傳遞和保存的方式都大為進步，使得人們可以利用數位資訊技術將重要訊息隱藏起來，因此現今有許多研究學者都致力於資訊隱藏的技術研究(Bender et al., 1995; Chun & Hsiang, 2002)，其中 Petitcolas 等人 (Petitcolas et al., 1999)根據資訊隱藏的功能和目的對資訊隱藏相關方面的技術做了分類，如圖 1 所示，其中資訊隱藏可以分成四大類，分別是隱蔽式通道(Covert Channels)，隱藏學(Steganography)，匿名法(Anonymity)及版權標記法(Copyright Marking)，而隱藏學又可分成兩類，分別是語言隱藏學(Linguistic Steganography)和技術隱藏學(Technical Steganography)，還有版權標記法也分成了強韌的版權標記和易碎性浮水印。還有版權標記法也分成了強韌的版權標記和易碎性浮水印。

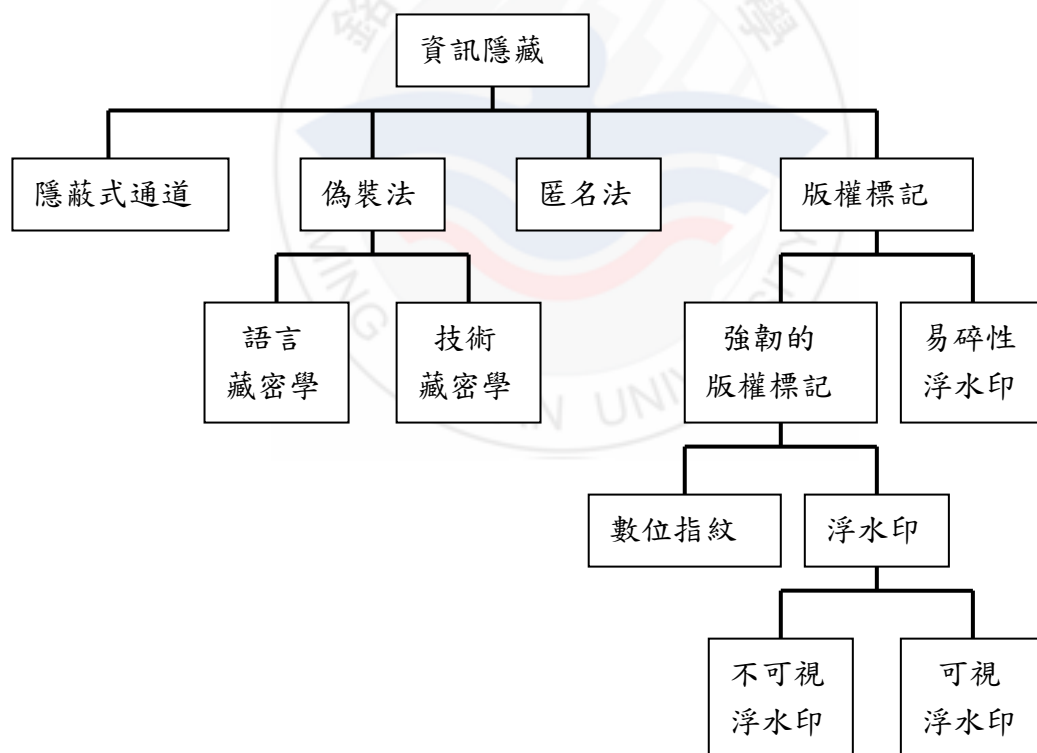


圖 1 資訊隱藏分類圖

而數位浮水印和隱藏學是目前在資訊隱藏領域中較為重要的技術。其中，數位浮水印主要是用來證明版權的所有，作法是將原創者的數位浮水印嵌入數位媒體中，並在經過數種攻擊或運算後還能萃取出相當程度的數位浮水印來證明版權的所有權。數位浮水印又可分為強韌性浮水印與易碎性浮水印，強韌性浮水印注重浮水印的抗攻擊性，越是能夠抵抗多種的攻擊並保持浮水印的完整性，則代表其強韌性越好；而易碎性浮水印則注重原始影像的完整性，其易碎性越高則代表越能夠偵測影像那裡被破壞或是經過竄改，達到原始影像的完整性；也有部分學者致力於浮水印協定(Chen & Tsai, 2006; Lei et al., 2004; Memon & Wong, 2001; Qiao & Nahrstedt, 1998; Zhang et al., 2006)上，更能提升版權的所有權。而資訊隱藏中的另一類：隱藏學，其所指的是將有意義的機密訊息藏入數位媒體中之後再傳遞，又可稱為「偽裝學」。隱藏學的主要目的在於將機密訊息隱藏在一張圖片或數位媒體中，藉由看似平常的傳遞行為，能夠將機密訊息一同傳遞到目的地，不被外界看穿有任何異常行為，既方便又安全。例如像是「視覺密碼」也算是隱藏學的一種，於 1995 年由 Naor 和 Shamir 所提出(Naor & Shamir, 1995)，視覺密碼指的是將一張秘密訊息影像利用一些公式或是方法將秘密訊息分解成二張或多張有意義或無意義的分享影像(Shares)並散佈出去，如果要知道秘密訊息，只有單張分享影像是無法得知秘密訊息的，而是需要將這些分享影像全部或部分重疊在一起，而且不需透過繁雜的數學運算，就可以用人類的視覺系統直接看到其中的秘密訊息，目前有許多學者致力於這方面的研究(Hou & Chen, 2000; Lin & Tsai, 2003; Lou et al., 2007; Wang et al., 2007)。「最低位元置換法」(Least Significant Bit Substitution, LSB Substitution) (Katzenbeisser & Petitcolas, 2000)也是隱藏學的一種，所謂的最低位元置換法指的是直接將秘密訊息分解後再藏入掩護影像(Cover Image)的每個像素值的最末幾個位元，通常是最末一個或最末兩個位元中，在盡量不破壞掩護影像的影像品質下，達到藏入秘密訊息的目的。

總而言之，不論是哪一種類型的資訊隱藏，都希望能夠做到盡量不破壞原始影像，這樣才能不被非法惡意者發現原圖中已藏有重要訊息。因此，現在有許多

學者都致力於資訊隱藏的研究，希望藉由提升資訊隱藏的技術水準以增加傳遞秘密訊息的安全性、保護數位媒體能不被惡意破壞、竄改或是能證明版權的所有。

1.2 研究動機

因為網際網路的技術蓬勃發展，使得資訊安全變的越來越重要，而資訊安全中的資訊隱藏是當中一個很重要的區塊，因此，近代有許多學者致力於研究資訊隱藏的相關技術，其中，有學者提出了最低位元置換法(Katzenbeisser & Petitcolas, 2000)的隱藏技術，可以用簡單的理論將秘密訊息藏入掩護影像中，但是卻產生了一個很大的問題，因為最低位元置換法的理論過於簡單和簡陋，而且被藏入秘密訊息的量越大，其偽裝影像(Stego Image)的影像品質會降低容易被識破，一旦被非法惡意者中途攔截，那麼當中的機密資訊很容易就被破解而取出，這樣安全性就大為降低了。因此，後來有其他學者提出了改良方法，像是 Wang 等人(Wang et al., 2001)在 2001 年提出了一個結合最佳顏色替代矩陣的最低位元置換法，可以改善偽裝影像的影像品質，因此能夠增加藏入秘密訊息的容量，又可增加機密訊息的安全性，但是有一個很大的缺點，就是會浪費太多時間在搜尋最佳顏色替代矩陣上。而後，Wang 等人(Wang et al., 2001)又提出了基因演算法求解顏色替代矩陣，以節省顏色替代矩陣的建構時間。基因演算法可分為二元編碼基因演算法(Binary GA)和實數編碼基因演算法(Real-parameter GA)。二元編碼演算法在每次求解問題時，都必須將原問題轉換為二元編碼的染色體(Chromosomes)才可進行演化，因此編碼的方法會影響求解的效率，在某些問題上，例如組合最佳化問題，染色體的二元編碼並不容易，經常會導致解答空間太大的問題；此外，太長的染色體字串長度也會導致較差的解答效率。而在實數編碼基因演算法方面，雖然可以縮短染色體編碼長度，但是其解答效率與品質並不會比二元編碼法好，所以不論是哪種基因演算法都有其限制與缺陷。根據 Dorigo 等人(Dorigo et al., 1996)及 Merkle 等人(Merkle et al., 2002)的研究得知：在求解旅行商人的問題(The Traveling Salesman Problem, TSP)的問題上，已證明蟻群最佳化演算法所求得的

解優於基因演算法與模擬退火法，而且當組合最佳化的問題越大時，基因演算法的執行效能越不如蟻群最佳化演算法。

因此，本論文提出一個結合蟻群最佳化演算法來求解替代矩陣的資訊隱藏法。蟻群最佳化演算法是在 1997 年由 Dorigo 等人(Dorigo & Thomas, 2004)所提出，其想法是觀察大自然中蟻群的特有行為而得到的啟發。在大自然中的真實螞蟻是藉由名為費洛蒙(Phermones)的一種化學物質來進行溝通，使得蟻群最後能夠以最短的時間找到品質更為優良的近似最佳食物解。蟻群最佳化演算法後來常被其他研究學者運用在求解最佳組合解方面，例如像是求解旅行商人的問題，因為蟻群最佳化演算法能夠降低時間複雜度求得品質更為優良的近似最佳解，而求解替代矩陣也是類似的問題，所以本論文提出了結合蟻群最佳化演算法去求解替代矩陣，希望可以花費較少的時間去求取一個較佳的替代矩陣，增加偽裝影像的影像品質。

一般而言，在傳統的 LSB 研究上只運用顏色替代矩陣來降低偽裝影像被破壞的程度，但是本研究認為只單獨運用顏色替代矩陣來藏入秘密訊息是不足夠，其降低偽裝影像被破壞的程度是有限的，因此，本文為了更增加安全性，而提出了應用蟻群最佳化演算法來建構結合的顏色替代矩陣與位置替代矩陣之 LSB 方法，更能夠降低偽裝影像被破壞的程度，提高安全性。因為，僅單獨運用顏色替代矩陣或位置替代矩陣只能稍微改善偽裝影像被破壞的程度，但如果結合使用，不僅能大大減少偽裝影像被破壞的程度，甚至能完全不破壞偽裝影像的影像品質。

再者，蟻群最佳化演算法未曾應用在影像處理方面，所以為了增加蟻群最佳化演算法的應用範圍，本論文特別結合了蟻群最佳化演算法去求解替代矩陣，希望可以多提供一個可應用蟻群最佳化演算法的新方向。

1.3 研究目的

不論是最原始的最低位元置換法或是運用最佳顏色替代矩陣的最低位元置換法都有存在一些缺點，像是偽裝影像的品質降低而使安全性降低或是增加了時間複雜度的問題，再者，顏色替代矩陣的效果有限，因此本論另外提出了位置替代矩陣，並將顏色替代矩陣與位置替代矩陣結合應用，能更加改善偽裝影像的影像品質，最後，目前尚未有人將蟻群最佳化演算法應用於資訊隱藏方面，因此，本研究提出了結合蟻群最佳化演算法去求解替代矩陣，希望可以藉由蟻群最佳化演算法所求得的替代矩陣提升偽裝影像的影像品質來增加安全性，或是減少嵌入秘密訊息的總體時間。

1.4 研究架構

本研究主要可以分成五個章節，第一章緒論中介紹研究背景包含資訊隱藏的重要性及分類，另外還有研究的動機及目的，第二章文獻討中介紹關於本研究應用的相關文獻及技術，其中包含最低位元置換法及其他的改良技術，還有蟻群最佳化演算法主要觀念，包含蟻群最佳化演算法的啟發或是如何應用及參數設定等觀念，第三章研究方法中介紹本研究的研究方法是如何進行，其中包含浮水印的嵌入方法及浮水印的偵測方法，而浮水印的嵌入方法中提到如何結合最低位元置換法與蟻群最佳化演算法並應用，是本研究的重心所在，第肆章實驗結果與分析中我們會呈現所得到的實驗結果，並互相作比較，第伍章結論中我們會介紹本研究的結論。

第貳章文獻探討

2.1 最低位元置換法 (Least Significant Bit Substitution)

能夠將秘密訊息藏入影像當中的一種技術。最初有簡單最低位元置換法 (Simple Least Significant Bit Substitution, Simple-LSB Substitution)，之後陸續有其他學者提出改良方法，或是結合不同的技術(Chan & Cheng, 2004; Thin & Lin, 2003)，使得藏入的秘密訊息安全度提升許多。

2.1.1 簡單最低位元置換法 (Simple-LSB Substitution)

簡單最低位元置換法(Katzenbeisser & Petitcolas, 2000)是最早被提出的一種資訊隱藏技術，它的方法簡單，實作上也容易執行。所謂的最低位元(Least Significant Bit)就是指像素(Pixel)中最不重要的部份，因為改變最低位元不會造成影像有太大的改變，肉眼上也不容易發現。因此，學者利用這個特性，將秘密資訊藏入像素值中的最後一個或最後兩個位元內，來達到資訊隱藏的目的。

例如有一張 8 位元的數位影像，其中三個像素值為 $(00100111)_2$ ， $(11101000)_2$ ， $(11001001)_2$ ，如果要將 $(100)_2$ 這一組秘密訊息藏入這三個像素中，則將秘密訊息拆成三組 1 位元的資訊分別藏入最後一個位元當中，藏入後像素值變成 $(00100111)_2$ ， $(11101000)_2$ ， $(11001000)_2$ ，就達到藏入訊息的目的了，如果要取出秘密訊息，只需要將最後一個位元取出就可以還原成原本的秘密訊息。

雖然藏入方法很簡單，取出也很方便，但是產生了兩個很大的問題，第一，如果秘密訊息越大，藏入原始影像當中所需的位元越多，則造成破壞原始的影像越明顯，這樣很容易讓竊取秘密資訊者發現這是一張偽裝影像；第二，因為這個方法很簡單，所以竊取秘密資訊者也能夠很容易發現秘密訊息並竊取它，因此在安全性方面還有很大的改進空間。因此，接下來有其他學者提出改良的方法，在下面幾節我們將介紹其他改良的方法。

2.1.2 最佳最低位元置換法

於 2001 年由 Wang 等人(Wang et al., 2001)所提出，這個方法主要是用於改善偽裝影像(Stego image)的影像品質和安全性。與簡單最低位元置換法相比較，主要增加了兩個不同的步驟(如圖 2 所示)，第一，秘密資訊 S 在拆解後變成 S' ，沒有立即藏入偽裝影像(host image)中，而是先利用混沌擾亂法(toral automorphisms) (Voyatzis & Pitas, 1996)做了加密的動作變成 S'' ，第二，秘密資訊加密後，接著計算一個能讓偽裝影像的信號雜訊比(Peak Signal To Noise Ratio, PSNR)最大的替代矩陣 $A = \{a_{ij}\}$ ，使 S'' 變成 S^* 之後，再藏進中最後得到偽裝影像 Z ，這樣就可以達到提昇影像品質的目的。

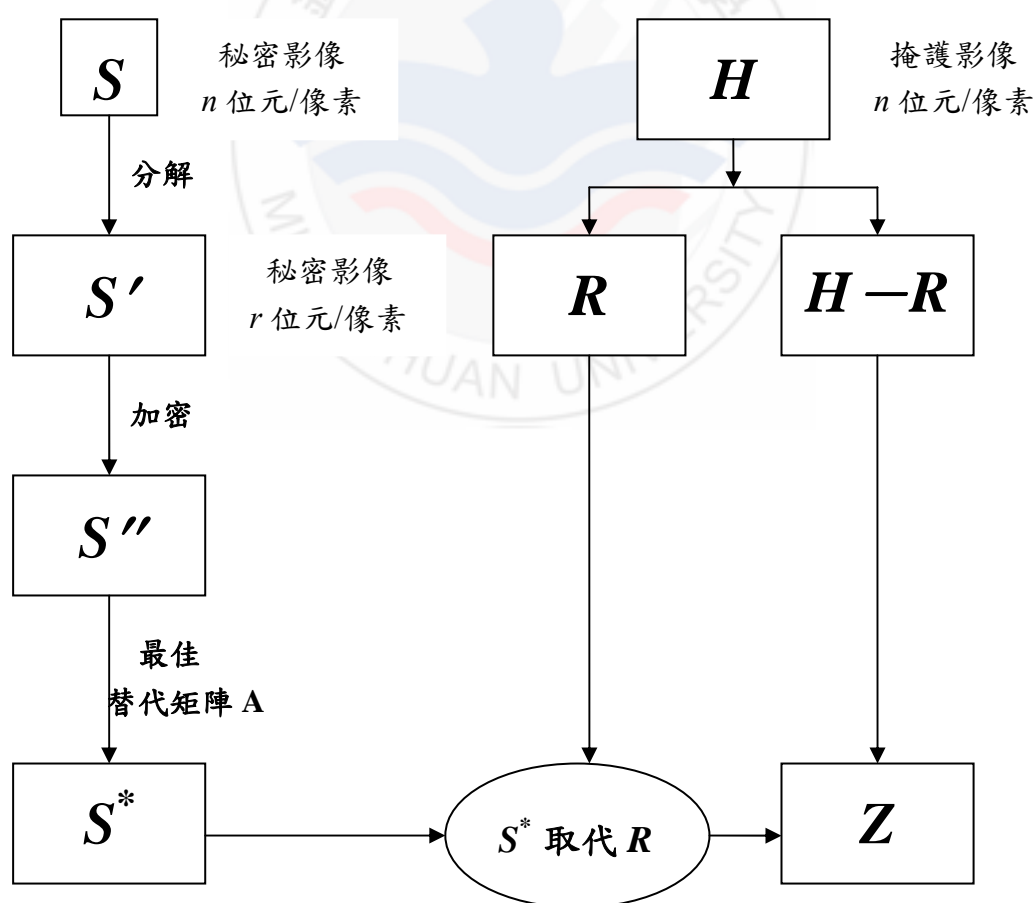


圖 2 最佳最低位元置換法的流程

加密的方法如公式(1)所示， k_0 和 k_1 是被用來當成鑰匙的兩個常數， x 是 S' 中每個像素值的位置， s 是 S 的影像大小，限制條件是 k_0 和 s 的最大公因數為 1，利用公式(1)打亂影像 S 的每個像素值的位置，一旦影像 Z 被人竊取，如果沒有 k_0 和 k_1 這兩把鑰匙也無法得知秘密資訊，所以可以藉由這種加密方法增加秘密資訊的安全性。

$$f(x) = (k_0 + k_1 \times x) \bmod s \quad (1)$$

第二步驟的替代矩陣 $A = \{a_{ij}\}$ (請參見公式(2))， ij 是替代矩陣的座標位置，每一行每一列都只能有一個 1。

$$a_{ij} = \begin{cases} 1 & \text{代表 } S' \text{ 影像中的 } i \text{ 值會被 } j \text{ 取代} \\ 0 & \text{不做任何改變} \end{cases} \quad (2)$$

但是有一件值得注意的事那就是：雖然最佳替代矩陣能夠降低偽裝影像 Z 的影像品質被破壞的程度，但是決定最佳替代矩陣是一件浩大的工程，假設每一個像素有 k 位元，就會有 $(2^k)!$ 種可能的替代矩陣，時間複雜度是呈指數關係成長的，因此 k 越大所產生可能的替代矩陣就越多。例如當 $k=4$ 時，則可行的顏色替代矩陣就有 $2^4! = 20,922,789,888,000$ 種，所以需要一套有效率的方法來產生適當的顏色替代矩陣。

2.1.3 利用基因演算法的最低位元置換法

雖然 Wang 等人(Wang et al., 2001)所提出的最佳最低位元置換法能夠提升秘密資訊的安全性和影像 Z 的品質，但是計算出最佳的替代矩陣所花費的時間多於簡單最低位元置換法所需時間許多，尤其當位元數 k 越大時，所需時間多出許多倍，因此 Wang 等人於 2001 年又提出了利用基因演算法(Genetic Algorithm) (Whitely, 1994)的最低位元置換法，以減少時間的支出。

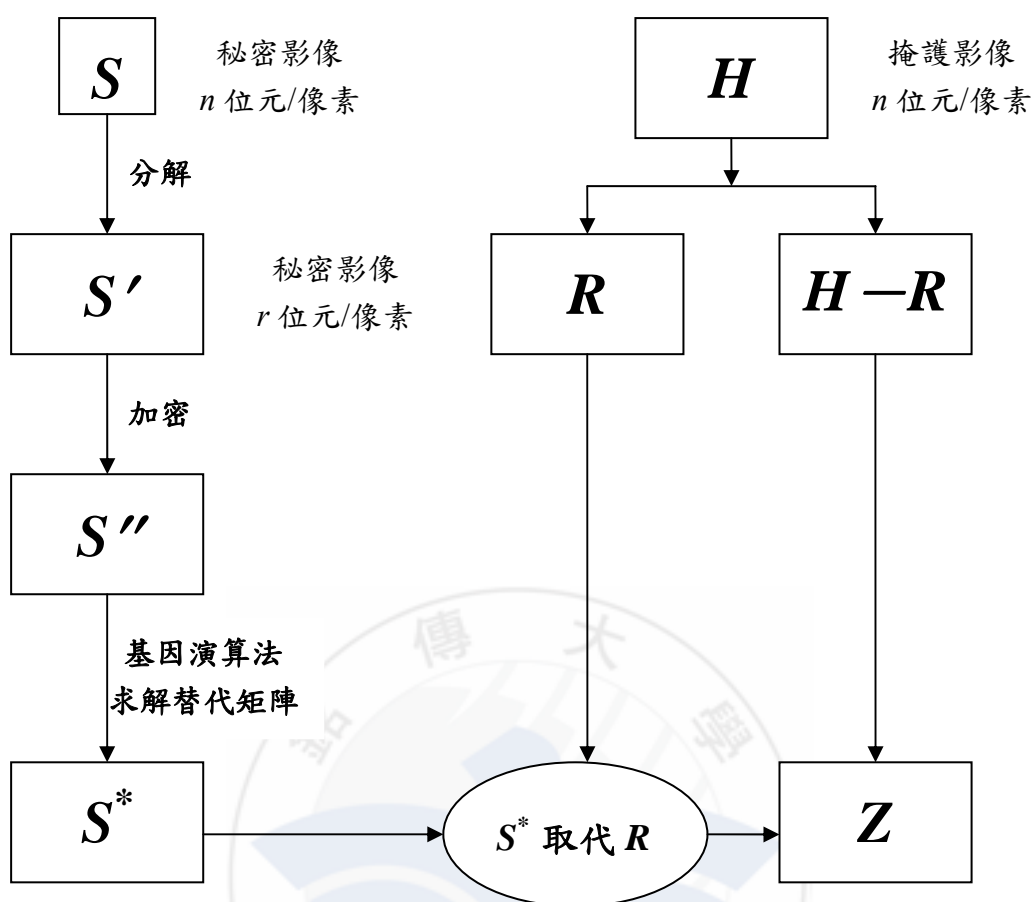


圖 3 基因演算法的最低位元置換法流程

基因演算法的最低位元置換法與最佳位元置換法的最大不同點在於求取替代矩陣的方式改為運用基因演算法(如圖 3 所示)。基因演算法主要有三個過程，分別是複製(Reproduction)、交配(Crossover)和突變(Mutation)，而在這篇文章中，作者主要是運用複製、交配和突變來求解替代矩陣。

我們利用圖 4 來說明基因演算法與替代矩陣之間的關係，圖 4 的上方是一個替代矩陣，在替代矩陣中，由上而下看到矩陣中分別是在座標位置(0,1)，(1,3)，(2,0)和(3,2)位置上的元素是 1，其餘都是 0。之後，取這些座標位置的行向量，可以得到分別是 1、3、0、2，所以作者就用 1、3、0、2 這四個數字代表一個染色體(Chromosome)中的四個基因，其中這四個基因是不能重複的。

		0	1	2	3
	0	0	1	0	0
	1	0	0	0	1
S	2	1	0	0	0
	3	0	0	1	0

G	1	3	0	2
-----	---	---	---	---

圖 4 基因演算法與替代矩陣關係圖解

接下來，我們介紹如何運用基因演算法的交配運算過程(利用圖 5)。如圖 5(a)所示， G_1 和 G_2 分別是兩個不同的染色體，分別有 8 個不同的基因，然後染色體 G_1 和 G_2 各自分裂成一半，前半部份保留，後半部分與對方交換，這就是所謂的交配，經過交配之後如圖 5(b)所示，接著如圖 5(c)所示，將對應每個不同的基因的箱子(box)先全部設定為「F」，接著從左到右掃描(Scan)兩個交配後的染色體中所有的基因，如果染色體當中的基因是已出現過的，就改變其基因值變成「-1」，而在上面的箱子中，將有掃描到的基因數字與其所對應的位置改為「N」，反之，則不改變，改變後的狀態如圖 5(d)所示，在圖 5(d)中，第一排數字是用來對應 box 的，第二排是 box，第三排是基因數字列，像在 G_2' 中，第一個掃描到的基因數字為 0，接著把第一排數字列為 0 的正下方的 box 中的「F」改成「N」，同樣的，掃描到的第二個基因數字為 2，就把第一排數字列為 2 的正下方的 box 中的「F」改成「N」，一直重複這樣的動作，直到整個染色體都被掃描完，掃描後發現對應第一排數字中的 1 和 3 在 box 中是「F」，代表染色體 G_2' 中目前沒有這兩種基因數字，接著再把這兩個基因數字分派到基因數字為「-1」的位置，最後可以得到圖 5(e)中的染色體 G_1' 和 G_2' 。

得到染色體 G_1' 和 G_2' 後，再利用適應函數決定哪一個基因會是較好的染色

體。這裡的適應函數是計算那一個染色體所轉變成的替代矩陣能使偽裝影像品質改善最多，經由上述步驟，我們就可以得到替代矩陣解了。

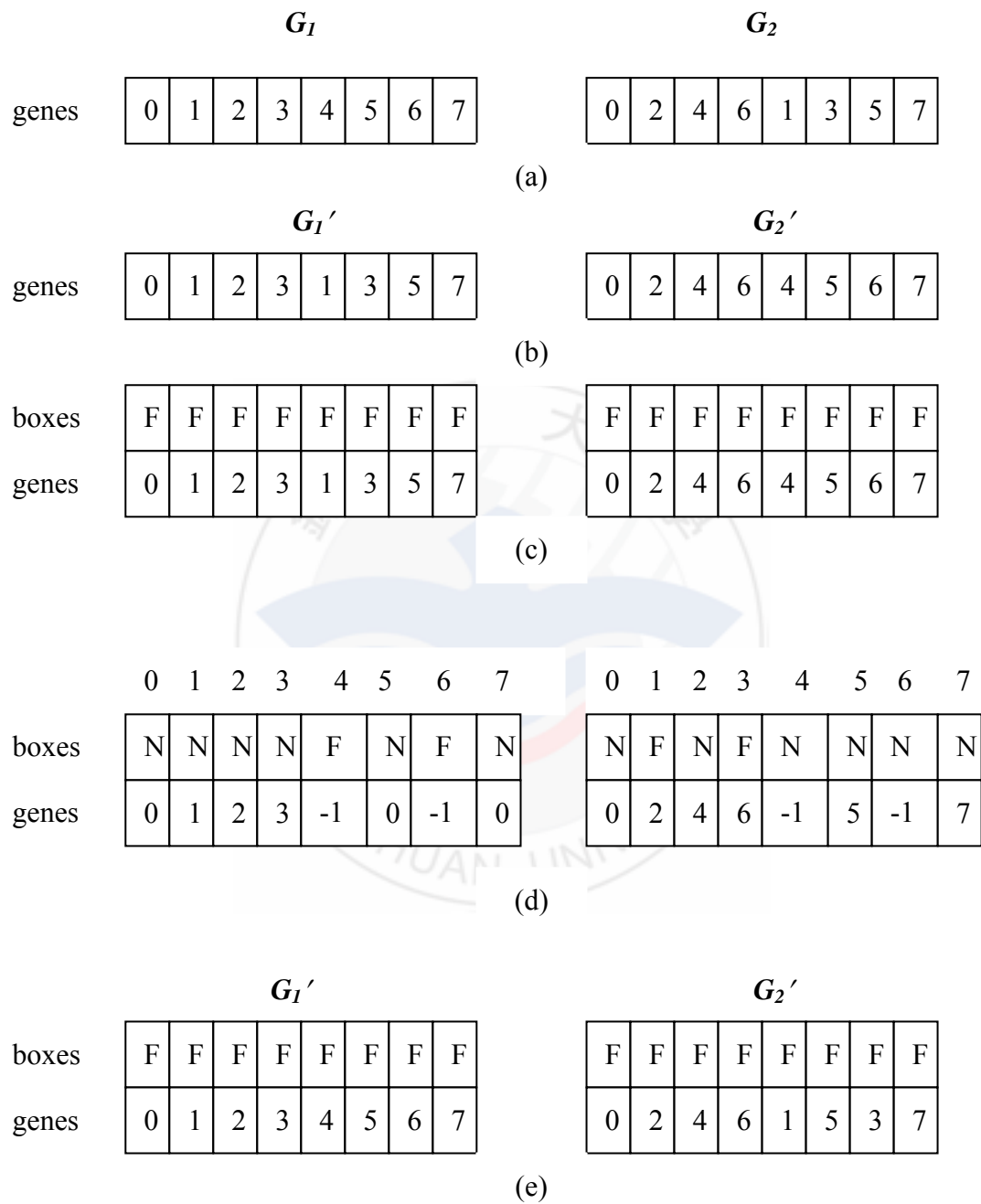


圖 5 基因演算法交配過程範例

2.1.4 基於 DES 和貪婪演算法的最低位元置換法

在 Wang 等人提出最佳最低位元置換法和基因演算法後，Chang 等人(Chang et al., 2002)在 2002 年所提出的基於 DES 和貪婪演算法的最低位元置換法是根據最佳最低位元置換法的缺點做改良。

Chang 等人所提出的方法與 Wang 所提出的最佳最低位元置換法主要有兩點不同(如圖 6 所示)，第一，將秘密資訊 S 加密變成的 S' 的方法改成 DES 加密法，第二，將加密後的秘密資訊 S'' 變成 S^* 的方法改成利用貪婪演算法來尋找替代矩陣，減少計算時間。

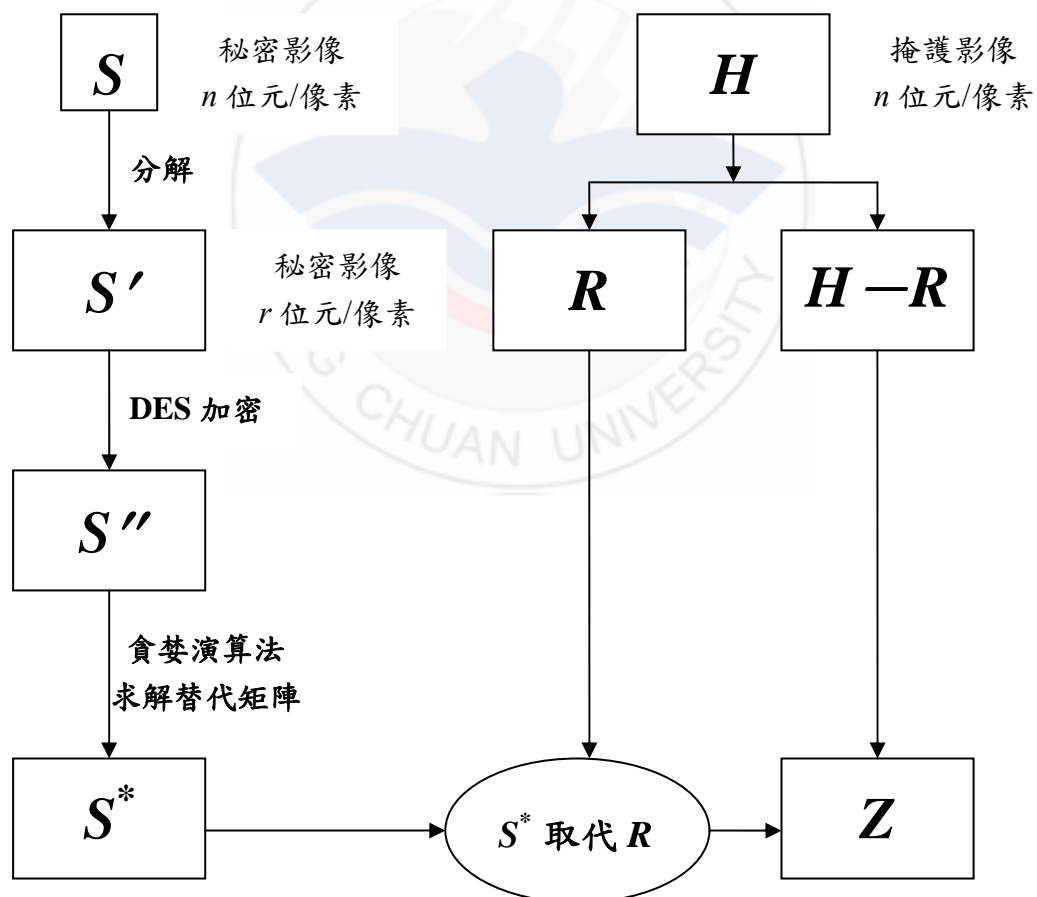


圖 6 基於 DES 和貪婪演算法的最低位元置換法流程圖

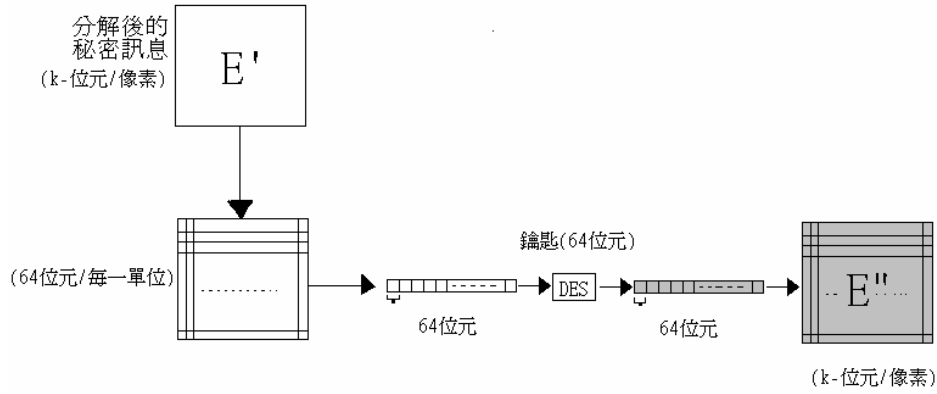


圖 7 DES 加密步驟

DES 是一種對稱的塊狀密碼器(Block Cipher)加密方法，假設每一個區塊的大小是 64 位元，則 DES 加密法也會產生 64 位元的鑰匙(如圖 7 所示)。在這步驟中，先將秘密資訊 S 從上至下從左至右裁剪成每一區塊為 64 位元，然後每一個區塊都分別利用 DES 加密法作加密，最後將加密過的區塊由上至下由左至右的集合在一起成了一張與 S 長寬都相同的影像 S'' 。

當秘密資訊經過加密後就使用貪婪演算法(Ahuja et al., 2000)尋找替代矩陣 $A = \{a_{ij}\}$ ，先利用加密後的秘密資訊 S'' 產生三個矩陣，分別是次數矩陣 (Occurrence Matrix) O ，權重矩陣 (Weight Matrix) W ，和成本矩陣 (Cost Matrix) C ，並且符合公式(3)及(4)，

$$w_{ij} = (i - j)^2 \quad (3)$$

$$c_{ij} = \sum_{j=0}^{2^{k-1}} o_{ij} w_{ij} \quad (4)$$

接著執行下面六個步驟：

步驟一：將替代矩陣 $A = \{a_{ij}\}$ 中的所有值都設為零。

步驟二：讀取成本矩陣第一列，從左至右找到最小的 c_{ij} ，如果找到兩個相同大

小的 c_{ij} 則選擇 j 較小的那一個，並刪除其他的 c_{ij} 。

步驟三：將相對應的 a_{ij} 在替代矩陣中設為 1。

步驟四：重複步驟二和步驟三，且規定每一行每一列只能有一個 a_{ij} 被設為 1，直到完成整個成本矩陣。

步驟五：重複步驟一到步驟四，並在成本矩陣第一列中找到不同的 c_{ij} ，以形成其他不同的替代矩陣。

步驟六：從之前計算出來的替代矩陣中，決定哪一個替代矩陣可以讓偽裝影像的影像品質最好。

步驟七：使用可以讓偽裝影像的影像品質最好的替代矩陣，完成藏入秘密訊息的動作。

基於 DES 和貪婪最低位元置換法所花費的時間比最佳最低位元置換法減少許多，但比簡單最低位元置換法的偽裝影像品質好許多，又可快速完成偽裝動作。

2.1.5 基於 JPEG 和粒子群演算法的最低位元置換法

由 Li 等人(Li & Wang, 2007)於 2007 年所提出，主要是改良 Upham 所提出的 Jpeg-Jsteg 方法(Upham)和 Chang 等人所提出的 JQTM 方法(Chang et al., 2002)。基於 JPEG 和粒子群演算法(Particle Swarm Optimization Algorithm, PSO Algorithm) (Kennedy & Eberhart, 1995)的最低位元置換法與之前幾節所提出的方法不太相同，之前幾節所提出的最低位元置換法是運用在空間域(spatial domain) (Zhang & Wang, 2005)，而此節所提及之方法主要是運用在頻率域(frequency domain) (Noda et al., 2006; Zhou et al., 2007; Huang et al., 2006; Tsai & Hung, 2004)。

其中，Upham 所提出的 Jpeg-Jsteg 方法步驟如下：

步驟一：利用 Jpeg-Jsteg 工具將一張掩護影像分割成一張 8×8 不重疊的區塊，再利用離散餘弦轉換(Discrete Fourier Transform, DFT)的公式轉換每一

個區塊，最後得到離散餘弦轉換的係數。

步驟二：將步驟一所得到的離散餘弦轉換係數根據標準 JPEG 量化表(如圖 8 所示)作衡量(Linde et al., 1980)，其衡量公式為公式(5)，衡量後可以得到量化過的離散餘弦轉換係數 $DCT^Q(X,Y)$ 。其中 $DCT(X,Y)$ 指的是座標位置在 (X,Y) 上的 DCT 係數， $Q(X,Y)$ 指的是座標位置在 (X,Y) 上的 JPEG 標準量化表中的數字。

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

圖 8 JPEG 標準量化表

$$DCT^Q(X,Y) = \text{Round} \left(\frac{DCT(X,Y)}{Q(X,Y)} \right) \quad (5)$$

步驟三：將要藏入的秘密訊息作編碼，再利用 LSB 方法，將編碼過的秘密訊息藏入量化過的離散餘弦係數 $DCT^Q(X,Y)$ 中非 0，1，或-1 之數值中，其藏入秘密訊息的順序如圖 9 的數字順序所示，如 Z 字型的順序藏入。

1	2	6	7	15	16	28	29
3	5	8	14	17	27	30	43
4	9	13	18	26	31	42	44
10	12	19	25	32	41	45	54
11	20	24	33	40	46	53	55
21	23	34	39	47	52	56	61
22	35	38	48	51	57	60	62
36	37	49	50	58	59	63	64

圖 9 藏入秘密訊息的順序

但是使用 Jpeg-Jsteg 方法，因為秘密訊息只能藏在 $DCT^Q(X,Y)$ 為非 1、-1 和 0 的數值中，所以能藏入秘密訊息的量非常少，因此，Chang 等人之後便提出了 JQTM 方法，目的是為了增加能夠藏入秘密訊息的容量。JQTM 方法主要改良了 JPEG 標準量化表中的數字，將量化表中中頻帶(Middle-frequency) 的數字改為 1，則 $DCT^Q(X,Y)$ 數值在中頻帶的部份就不為 1、-1 和 0，這樣就符合可以藏入秘密訊息的條件，使秘密訊息全部藏入中頻帶的部份，因此利用 JQTM 方法就可以增加能夠藏入秘密訊息的容量。其中，JQTM 方法改變後的 JPEG 量化表，如圖 10 所示，在中頻帶的部份，共有 26 個數字被設為 1。

16	11	10	16	1	1	1	1
12	12	14	1	1	1	1	55
14	13	1	1	1	1	69	56
14	1	1	1	1	87	80	62
1	1	1	1	68	109	103	77
1	1	1	64	81	104	113	92
1	1	78	87	103	121	120	101
1	92	95	98	112	100	103	99

圖 10 JQTM 方法的量化表

雖然使用 Chang 等人所提出之 JQTM 方法可以增加藏匿秘密訊息的容量，但是若秘密訊息更為龐大，則可以藏匿的容量還是不太足夠，因此，Li 等人於 2007 年提出了基於 JPEG 和粒子群演算法的最低位元置換法改良藏匿容量不足的問題，更利用替代矩陣改善影像品質增加安全性。其嵌入秘密訊息的過程如圖 11 所示，其中，秘密訊息的部份是先利用粒子群演算法算出替代矩陣，再利用替代矩陣改變秘密訊息的像素值，使秘密訊息遷入掩護影像時，能夠盡量不破壞掩護影像的影像品質，增加秘密訊息的安全性。而在掩護影像的部份，與 Jpeg-Jsteg 方法相似，先將掩護影像分割成一張 8×8 不重疊的區塊，再利用離散餘弦轉換的公式分別轉換每一個區塊，最後得到離散餘弦轉換的係數，在利用 Li 等人所提出的量化表求得量化後的數值，再將利用替代矩陣轉換過後的秘密訊息運用 LSB 方式嵌入量化後的數值中，最後得到一張已經嵌入秘密訊息的偽

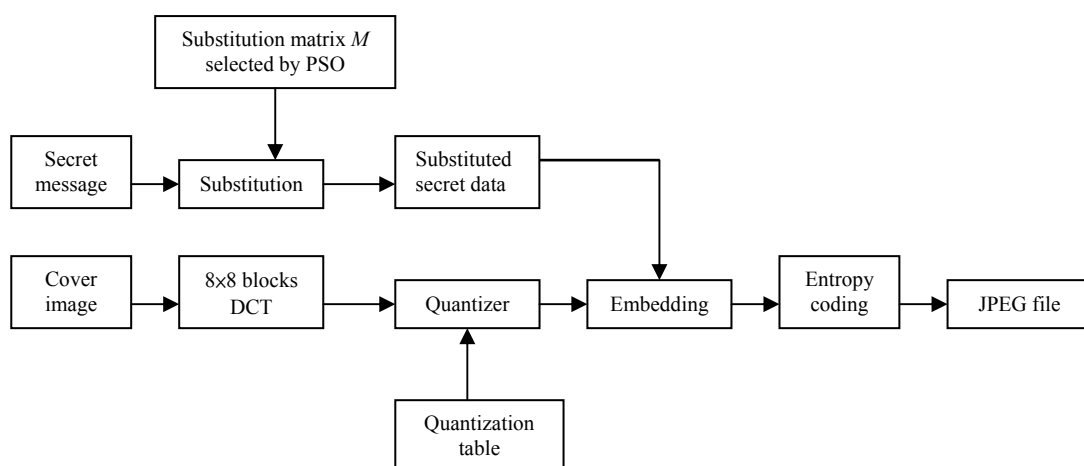


圖 11 基於 JPEG 和粒子群演算法的最低位元置換法嵌入秘密訊息流程裝影像。

其中，Li 等人所提議的量化表如圖 12 所示，最左上角的數值改成 8，低頻 (low frequency) 和中頻的都改為 1，其能夠藏入秘密訊息的部份為左上角到中頻的部份，比 JQTM 方法能夠藏匿秘密訊息的容量增加許多。

8	1	1	1	1	1	1	1
1	1	1	1	1	1	1	55
1	1	1	1	1	1	69	56
1	1	1	1	1	87	80	62
1	1	1	1	68	109	103	77
1	1	1	64	81	104	113	92
1	1	78	87	103	121	120	101
1	92	95	98	112	100	103	99

圖 12 基於 JPEG 和粒子群演算法的最低位元置換法所提議的量化表

在求算替代矩陣方面，是使用粒子群演算法來求得替代矩陣，其公式如公式 (6) (7)(8) 所示，

$$V_i = w \times V_i + c_1 \times rand_1 \times (pbest_i - X_i) + c_2 \times rand_2 \times (gbest_i - X_i) \quad (6)$$

$$X_i(t) = X_i(t-1) + V_i(t) \quad (7)$$

$$w = w_{\max} - n \times \frac{w_{\max} - w_{\min}}{\text{iter_max}} \quad (8)$$

其中， V_i 是第 i 個粒子的速度，也可說是第 i 個粒子的移動量， w 是權重， c_1 和 c_2 是用來加速作用的常數， rand_1 和 rand_2 是 0 到 1 之間的隨機亂數，可以避免最後得到的解是局部是最佳解， $pbest_i$ 是第 i 個粒子目前最佳的解， $gbest_i$ 是所有粒子中目前得到的最佳解， X_i 是第 i 個粒子目前所在位置， n 是目前反覆搜尋的次數， w_{\max} 和 w_{\min} 分別是權重的上限值和下限值， iter_max 是設定反覆搜尋的最大次數。

而求算替代矩陣的步驟如下：

步驟一： 將所有數值初始化，並隨機產生 K 個粒子，其所在位置是 X_i ，

$i=1,2,\dots,K$ ，其速度是 V_i ，並設定 $pbest_i = X_i$ 。

步驟二： 更新粒子的速度和位置，利用公式(6) (7)(8)去更新粒子的速度和位置，

並計算每次每個粒子的 $pbest$ 和每次的 $gbest_i$ 。

步驟三： 求得 $gbest_i$ 之後與前一次的解答相比，如果得到的 PSNR 值較大，則

保留新求得的 $gbest_i$ ，反之，則保留舊的解答。

步驟四： 重複執行步驟二和步驟三，直到 $n=\text{iter_max}$ 為止。

由於粒子群演算法所求得的答案不一定為整數或是不一定能符合運用在替代矩陣上，所以需要做一些小改變，如得到的解 $X_i=(-0.2, 2.5, 4.1, 1.3)$ ，則需將解答更改為(0, 2, 3, 1)，這樣就能符合替代矩陣要的答案，其解答對應到替代矩陣上的方式與基因演算法的方式相同。

2.2 蟻群最佳化(Ant Colony Optimization)方法之演進

蟻群最佳化方法最早是由 Dorigo 於 1991 年提出(Dorigo and Thomas, 2004)，是根據大自然界螞蟻的行為得到啟發，現在已經能用來求解最佳化組合的問題。接下來，我們將一一介紹蟻群最佳化方法的由來和演進。

2.2.1 自然界螞蟻

Dorigo 在觀察大自然時，發現螞蟻能夠不透過對話溝通就找到之前的蟻群所走過的路徑，經過研究後發現，螞蟻是藉由一種化學物質的氣味而找到之前的蟻群所走過的路徑，這種化學物質稱為費洛蒙(Pheromones)。一旦螞蟻跟著留有費洛蒙氣味的路徑行走，便可找到最佳化路徑或最短路徑。

為了更清楚的表達螞蟻的行走路徑的選擇，我們用圖 13(a)和圖 13(b)來解釋。在圖 13(a)中，螞蟻的巢穴到食物的路徑有兩條，上下兩條路徑的長短皆相同。此時若有兩隻螞蟻同時從巢穴出發，有 50%的機率會選擇路徑一，也有 50%的機率會選擇路徑二。由於兩條路徑的長度相同，假設兩隻螞蟻的速率相同，則兩隻螞蟻將同時到達食物區。當螞蟻準備回巢穴時，它們所面對的兩條路徑都是只有一份費洛蒙痕跡(Trail Pheromone)，因此回巢穴的兩條路徑機率也都是相同的，各為 50%。

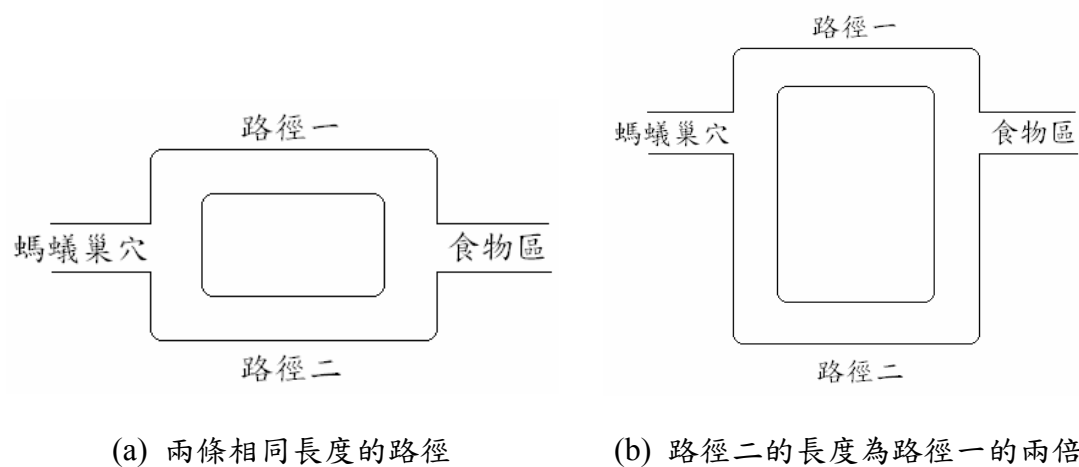


圖 13 巢穴到食物區的路徑圖

相反的，若路徑的長度不同，螞蟻將選擇較短的路徑行走，如在圖 13(b)中，螞蟻的巢穴到食物的路徑有兩條，路徑二的長度約為路徑一長度的兩倍。此時若有兩隻螞蟻 A、B，同時從巢穴出發，有 50%的機率會選擇上面的路徑，也有 50%的機率會選擇下面的路徑。由於兩條路徑的長度不同(假設速率相同)，則選擇路徑一的螞蟻將先到達食物區。當先到達的螞蟻 A 欲回巢穴時，面對了兩條路徑，一條路徑是已有一份費洛蒙痕跡的路徑一，而另一條完全沒有費洛蒙痕跡。無庸置疑的，螞蟻將會選擇有費洛蒙痕跡的路徑一(較短路徑)回到巢穴。此時另一隻螞蟻 B 也到達了食物區，當欲回巢穴時，面對的兩條路徑，路徑一是有兩份費洛蒙痕跡，而路徑二只有一份費洛蒙痕跡。所以螞蟻 B 則選擇路徑一(較短路徑)回巢穴。

2.2.2 人工螞蟻

Dorigo 在觀察大自然界螞蟻後，為了能夠更深入了解並且研究螞蟻所選擇的路徑，因此，Dorigo 創造了人工螞蟻(Artificial Ants)代替大自然界的螞蟻進行實驗。Dorigo 最初所做的實驗得到的演算法稱為簡單蟻群最佳化演算法(Simple-Ant Colony Optimization, S-ACO)。接下來我們介紹簡單蟻群最佳化演算法利用人工螞蟻所得到的定義過程和結論。人工螞蟻是根據真實螞蟻的特徵所做出，仍然是依據費洛蒙的濃度，判斷欲行走的路徑。意指路徑的費洛蒙濃度越高則行走的機率越大，並可藉此尋找到最短路徑。但人工螞蟻與真實螞蟻仍然有不同之處：

- 1.螞蟻擁有記憶路徑的功能。當到達目的地欲回巢穴時，螞蟻可藉由記憶功能避開迴圈，並在回程途中增加費洛蒙供給其他螞蟻使用。
- 2.為了避免產生迴圈，蟻群會利用之前其他螞蟻所遺留的費洛蒙來決定行經路線，並且前往食物區的途中不留下任何費洛蒙。
- 3.蟻群在回巢穴途中，會根據食物區的食物品質好壞來決定該增加多少費洛蒙的

濃度，以幫助其他螞蟻能行走最短路徑並找到品質最好的食物。費洛蒙增加公式如下：

$$\tau_{ij} \leftarrow \tau_{ij} + \Delta\tau^k, \quad (9)$$

其中 τ_{ij} 代表路徑 i 到 j 之間的費洛蒙濃度，且 $\Delta\tau^k$ 代表第 k 隻螞蟻經過路徑 ij 所增加的費洛蒙濃度

4.可以藉由調整費洛蒙蒸發(Pheromone Evaporation)速率，以避免得到品質較差的路徑解。費洛蒙蒸發公式如下：

$$\tau_{ij} \leftarrow (1 - \rho)\tau_{ij}, \quad (10)$$

其中 τ_{ij} 代表路徑 i 到 j 之間的費洛蒙濃度，且 ρ 代表一個包含 0 但小於 1 的一個參數。 ρ 值如果越大，則代表費洛蒙蒸發速率越快；相反的， ρ 值越小，則代表蒸發速率越慢。

5.我們可以依據費洛蒙濃度計算下一個應該要選取的位置，其機率公式為：

$$P_{ij}^k = \begin{cases} \frac{\tau_{ij}^\alpha}{\sum_{l \in N_i^k} \tau_{il}^\alpha}, & \text{if } j \in N_i^k, \\ 0, & \text{if } j \notin N_i^k. \end{cases} \quad (11)$$

其中 P_{ij}^k 指的是選擇下一個位置 j 的機率， α 指的是一個調整參數， N_i^k 指的是螞蟻 k 在位置 i 時，鄰近的所有的點的集合。若下一個位置 j 不屬於 N_i^k ，則此時選擇 j 的機率為零。在這個公式中我們知道利用編號為 k 的人工螞蟻，在位置 i 時，究竟要行走哪一條路徑，就是根據 P_{ij}^k 來決定。

2.2.3 蟻群最佳化次經驗(ACO-Metaheuristic)

組合最佳化問題的解決辦法可分成下列兩種：

1. 精確的演算法(Exact Algorithms)：指的是運用暴力法(Brute Force)去慢慢的尋找精確的解答，也就是所謂的土法煉鋼法，完全沒有任何技巧，只是一步步的慢慢的去尋找解答，因此往往解決一個問題都需要大量的時間和金錢，尤其當問題的模型越大，解決精確的答案更是一件不可能的任務。
2. 近似演算法(Approximate Algorithms)：指的是為了避免繁雜的運算和花費大量的時間與金錢，而演化得來的一種演算法。雖然省去了大量的計算時間，但所得解並不保證為精確最佳解。

近似演算法又可分類為下列兩種演算法：

1. 推測演算法(Constructive Algorithms)：

推測演算法所指的是剛開始只有一個解，慢慢的反覆一個一個增加其他解答。增加的速度可以非常快。如圖 14 所示，原本只知道 a，但後來一個一個慢慢增加，然後增加 b，接著增加 c，d，e。最後整個圖形架構都被找尋到了。

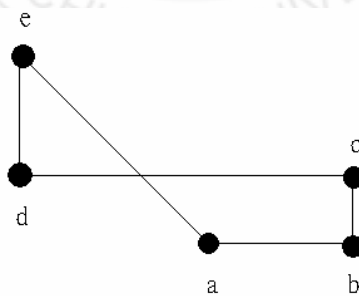


圖 14 推測演算法的圖解

2. 局部搜尋法(Local Search)：

局部搜尋法指的是藉著反覆的在鄰近地區搜尋其他尚未找尋過的鄰近解，以試圖改善現有的答案。不論是推測演算法或是局部搜尋法所找尋到的解答通常都無法幫助找到具有重大改善的解，常常只得到有限數量的不同解

或是找到品質不佳的局部最佳解。因此，為了改善上述兩種方法的缺點，近代有學者發展出了新的最佳化問題的求解方法，稱為次經驗法(Metaheuristic)。次經驗法可以視為指導”啟發方法(Heuristic Method)”能夠朝向有希望去搜尋到更高品質的解的方向。次經驗法是種能夠增加能力去找到更高品質解的方法，甚至幾乎所有有關於組合最佳化的相關問題都能在合理的時間內找到解答。

相同的，蟻群最佳化次經驗(ACO-Metaheuristic)也能夠應用於許多靜態和動態的組合最佳化的相關問題上。例如：組合最佳化問題的靜態問題如旅行商人的問題(Traveling Salesman Problem)，組合最佳化問題的動態問題如網路選擇路由的問題(Network Routing Problem)。

如圖 15 所示為蟻群最佳化次經驗的虛擬碼，其中 ConstructAntsSolution 指的是蟻群同時和非同時的到達各個節點，並開始建立答案去找尋最佳解。UpdatePheromones 指的是更新費洛蒙濃度，包含增加費洛蒙濃度和費洛蒙濃度蒸發的行為。增加費洛蒙濃度可以幫助未來要行走此路徑的螞蟻可以迅速的找到最佳解，而費洛蒙蒸發可以避免蟻群迅速的收斂到一個子最佳解(Suboptimal)區域，並有利於蟻群去探索(Exploration)新的區域，增加找到全域最佳解(Global Optimal Solution)的機率。DaemonActions 指的是完成非單一隻螞蟻能夠完成的集

```
Procedure ACOMetaheuristic  
  ScheduleActivities  
    ConstrucAntSolutions  
    UpdatePheromones  
    DaemonActions      % optional  
  end—Schedule Activities
```

圖 15 蟻群最佳化次經驗的虛擬程式碼

中行為。需將所有螞蟻行走後所得到的解集中在一起，並判斷哪些解有利用價值，哪些解沒有，並不再增加費洛蒙濃度。

2.2.4 蟻群最佳化演算法

有關於蟻群最佳化演算法的第一個演算法稱為“螞蟻系統(Ant System)”，接著也有許多的關於蟻群最佳化演算法被提出。接下來，我們詳細介紹螞蟻系統的解答過程。最初，關於螞蟻系統有三種不同的看法，分別是螞蟻密度(Ant-density)，螞蟻數量(Ant-quantity)和螞蟻週期(Ant-cycle)。螞蟻密度和螞蟻數量這兩種看法類似，強調螞蟻從一個點移動到相鄰的另一個點之後，會立刻直接更新費洛蒙濃度。而螞蟻週期的看法則是在每隻螞蟻都各自建構完成自己的旅程圖後，才根據旅程的品質好壞決定應增加或減少費洛蒙的濃度，達成費洛蒙更新的行為。螞蟻系統演算法主要由旅程的建立(Tour Construction)及更新費洛蒙痕跡(Update Of Pheromone Trails)兩個部分構成。

首先，我們先介紹旅程的建立(Tour Construction)：在螞蟻系統中，所有的人工螞蟻會同時的建立自己的旅程。剛開始會隨機的選擇一個點開始行進，之後會利用機率的行為選擇法則，稱之為“隨機的比列法則(Random Proportional Rule)”，來決定螞蟻將要行走的下一個點。公式如下：

$$p_{ij}^k = \frac{[\tau_{ij}]^\alpha [\eta_{ij}]^\beta}{\sum_{l \in N_i^k} [\tau_{il}]^\alpha [\eta_{il}]^\beta}, \text{ if } j \in N_i^k, \quad (12)$$

其中 $\eta_{ij} = \frac{1}{d_{ij}}$ ， p_{ij}^k 是螞蟻 k 在點 i 決定要到點 j 的機率大小， τ_{ij} 是點 i 到點 j 路徑中的費洛蒙濃度， η_{ij} 是點 i 到點 j 的啟發值(Heuristic Value)， d_{ij} 是路徑長度， N_i^k 是螞蟻 k 在點 i 時，附近周圍的可行鄰近點，而且此鄰近點為尚未行經過的點；且 α 與 β 為兩個可調整的參數。在這公式當中，如果 α 與 β 設置不當，會產生較差的解。例如，若 $\alpha=0$ ，則將只選取最靠近 i 的點 ($\because \eta_{ij}=1/d_{ij}$)；若 $\beta=0$ ，

則選擇下一個點的機率完全只能依費洛蒙的濃度來做決定。在螞蟻系統當中，每一隻螞蟻都具有記憶功能，包含記憶那些點已經行經過(判斷可行鄰近解)，並能夠計算路徑的長度，最後靠著回憶路徑回巢穴並放置費洛蒙。

其次要介紹的是更新費洛蒙痕跡(Update Of Pheromone Trails)在所有螞蟻都建構完成自己的旅程後，才開始在回程更新費洛蒙濃度，意指「全域更新(Global Updating)」，費洛蒙蒸發公式如下：

$$\tau_{ij} \leftarrow (1 - \rho) \tau_{ij}, \quad \forall (i, j) \in L \quad (13)$$

其中 ρ 為大於零小於等於 1 的費洛蒙蒸發率， L 為路徑上的所有點集合。參數 ρ 的值越大，費洛蒙蒸發速率越快，可以幫助螞蟻忘記之前所選擇的不好的路徑。相反的，參數 ρ 的值越小，費洛蒙蒸發速率越慢，螞蟻就會容易陷入在之前所選擇的不好的路徑或是迴圈中。費洛蒙濃度增加公式如下：

$$\tau_{ij} \leftarrow \tau_{ij} + \sum_{k=1}^m \Delta \tau_{ij}^k, \quad \forall (i, j) \in L \quad (14)$$

其中 τ_{ij}^k 為螞蟻 k 在點 i 到點 j 路徑上所放置的費洛蒙痕跡，定義如下：

$$\Delta \tau_{ij}^k = \begin{cases} 1/C^k, & \text{if arc } (i, j) \text{ belongs to } T^k, \\ 0, & \text{otherwise.} \end{cases} \quad (15)$$

其中 T^k 是第 k 隻螞蟻所建立的旅程； C^k 是旅程 T^k 的長度，費洛蒙痕跡的增加與路徑長短成反比。因此，由上述得知，利用費洛蒙濃度多寡，可以讓蟻群快速的尋找到最佳路徑。

之後，Dorigo 又發展出延伸螞蟻系統的演算法，稱為蟻群系統(Ant Colony System)，也是蟻群最佳化演算法其中一種，主要差異在於蟻群系統比螞蟻系統更積極的利用累積的經驗去搜尋新解答，而且費洛更新及蒸發只應用在目前所找到路徑中的最佳路徑，其中費洛蒙更新可分成局部更新(Local Updating)和全域更

新兩種，讓蟻群可以在更短的時間中找到更佳の解。

2.2.5 蟻群最佳化演算法目前應用的範圍

1.旅行商人的問題(The Traveling Salesman Problem)

這個問題在描述一個國家或地區中有許多城市，城市間有道路相互連接，路徑有長短之分，這時，有一位旅行商人要完成任務，其任務就是：從某一個城市開始出發，途中需要經過每一個城市，而且只能一次，並求得最短路徑，所以需要應用求解最佳化的演算法來求解答案。

2.連續順序問題(The Sequential Ordering Problem)

在一個圖型中，各點可連成線，而且某些點之間具有特定之順序，起初，從某一點開始，需要每一點都經過而且只能一次，並符合其特定順序的最佳化問題。

3.廣義の指派問題(The Generalized Assignment Problem)

有 x 個任務要分派給 x 個代理人，其中每個代理人和任務都有各自的限制，最終目標是在符合限制下求得最小成本。

4.複合の背包問題(The Multiple Knapsack Problem)

假設全部共有 N 個項目，需從這 N 個項目中，適當的選取 n 個項目裝進有限容量的 M 個背包中，並且需要將 N 個項目全部分配裝進不同的背包中，最後求解每個背包之總和利潤極大。相當於將現有的資源分配給不同的專案，並求得專案的最终利潤的極大化。

5.網路路由問題(The Network Routing Problem)

網路路由問題著重在速度快、時間短，所以最終目的是求解網路節點中出發點與目的地的最短路徑。

6.動態的旅行商人問題(The Dynamic Traveling Salesman Problem)

與旅行商人問題大同小異，其最大不同點在於，動態旅行商人問題中的城市節點會突然增加或減少。

第參章研究方法

3.1 問題描述

資訊隱藏在資訊安全方面，是一個重要的研究議題，而所謂的資訊隱藏指的是：將有意義的資訊或影像藏入另一重要的影像或資訊中當成版權的標記，或是將有意義的資訊或影像藏入一個僅僅用來掩護的有意義的影像或資訊中。前者所指的有意義的影像或資訊在此被稱為浮水印，其存在的主要目的是為了證明版權的所有或是保持原始影像的完整性；後者所指的有意義的影像或資訊指的是真正的秘密訊息，這個方法稱為偽裝法(Steganography)，而偽裝法的主要目的在於將機密訊息隱藏在一張圖片中，藉由看似平常的傳遞圖片的行為，能夠將機密訊息一同傳遞到目的地，不被外界看穿有任何異常行為，既方便又安全。

一般而言，不論是哪一種類型的資訊隱藏，都希望能夠做到盡量不破壞原始影像為主要目標，這樣才能不被有心人士發現原圖中已藏有重要訊息。因此，近年來有學者提出最低位元置換法(Least Significant Bit Substitution, LSB Substitution)，藉由分解秘密訊息並藏入原圖的每個像素值的最後幾個位元，達到資訊隱藏的目的。但是，慢慢的這個方法變的不太安全，因為雖然只是改變原圖的最後幾個位元，但還是會被人察覺到原圖已經被更改，而且藏入秘密訊息的方法太過簡單，易被破解，因此，Wang 等人(Wang et al., 2001)提出了一個結合最佳顏色替代矩陣(Substitution Matrix)的最低位元置換法，可以改善偽裝影像的影像品質，既能夠增加藏入秘密訊息的容量，又可增加機密訊息的安全性，但其缺點則是需要花費許多時間在搜尋最佳顏色替代矩陣上。因此，Wang 等人(Wang et al., 2001)又提出以基因演算法(Genetic Algorithms)(Holland, 1975)來求解顏色替代矩陣的方法，以節省顏色替代矩陣的建構時間。

基因演算法可分為二元編碼基因演算法(Binary GA)和實數編碼基因演算法(Real-parameter GA)。二元編碼演算法在每次求解問題時，都必須將原問題轉換

為二元編碼的染色體(Chromosomes)才可進行演化，而編碼的方法則會影響求解的效率，在有些問題上，例如組合最佳化問題，染色體的二元編碼並不容易，經常會導致解答空間太大的問題；此外，太長的染色體字串長度也會導致較差的解答效率。而在實數編碼基因演算法方面，雖然可以縮短染色體編碼長度，但是其解答效率與品質並不會比二元編碼法好，所以不論是哪種基因演算法都有其限制與缺陷。根據 Dorigo 等人及 Merkle 等人的研究，在求解 TSP 的問題上，已證明蟻群最佳化演算法所求得的解優於基因演算法與模擬退火法；因此，本文提出了一個應用蟻群最佳化演算法來建構 LSB 像顏色替代矩陣之方法，使求解顏色替代矩陣的過程更加有效率。其中所使用的蟻群最佳化演算法大部分學者都使用在求解最佳組合解的問題上，例如像是旅行商人的問題上，因此，本論文希望能夠藉著結合蟻群最佳化演算法，花費較少的時間求得較好的解，也就是較好的替代矩陣。

但是，只單獨運用顏色替代矩陣來藏入秘密訊息是不足夠，只能稍微降低偽裝影像被破壞的程度。因此，本文為了更增加安全性，而提出了應用蟻群最佳化演算法來建構結合的顏色替代矩陣與位置替代矩陣之 LSB 方法，更能夠降低偽裝影像被破壞的程度，提高安全性。因為，僅單獨運用顏色替代矩陣或位置替代矩陣只能稍微改善偽裝影像被破壞的程度，但如果結合使用，不僅能大大減少偽裝影像被破壞的程度，甚至能完全不破壞掩護影像的影像品質就達到隱藏的目的。

3.2 符號定義

H : 大小為 $M \times N$ 的一張掩護影像(Cover Image)，用以藏入秘密訊息，其中每個像素的像素值為 n 個位元。

R : 大小為等於或小於 $M \times N$ 的一張影像，由掩護影像 H 分解而得，而且將會被秘密訊息所替代，其中每個像素的像素值為 r 位元。

$H-R$: 大小為等於或小於 $M \times N$ 的一張影像，由掩護影像 H 分解而得，其中每

個像素的像素值為 $n-r$ 位元。

Z : 大小為 $M \times N$ 的一張的偽裝影像(Stego Image)，其中每個像素的像素值為 n 位元。

S : 大小為 $m_0 \times n_0$ 的一個秘密訊息，其中每個像素的像素值為 r_0 位元。

S' : 大小為 $m \times n$ ($m \leq M, n \leq N$) 的一個秘密訊息，由 S 分解之後而得，其中每個像素為 r 個位元，且 $r < n$ ， $m \times n \times r = m_0 \times n_0 \times r_0$ 。

S'' : 大小為 $m \times n$ 的一個秘密訊息，由 S' 加密之後而得，其中每個像素的像素值為 r 個位元。

S^* : 大小為等於或小於 $m \times n$ 的一個秘密訊息，先將其分成大小相同且不重疊的 d 個區塊，再根據替代矩陣 A 及 B 改變 S'' 部份像素值及其區塊位置之後而得，其中每個像素的像素值為 r 位元。

A : 大小為 $\lambda \times \lambda$ 的一個方陣 $[a_{ij}]$ ，其中 i 表示方陣中的列， j 表示方陣中的行且 $\lambda = 2^r$ 。

B : 大小為 $\mu \times \mu$ 的一個方陣 $[b_{ij}]$ ，其中 i 表示方陣中的列， j 表示方陣中的行且 $\mu = d$ 。

3.3 秘密訊息的隱藏

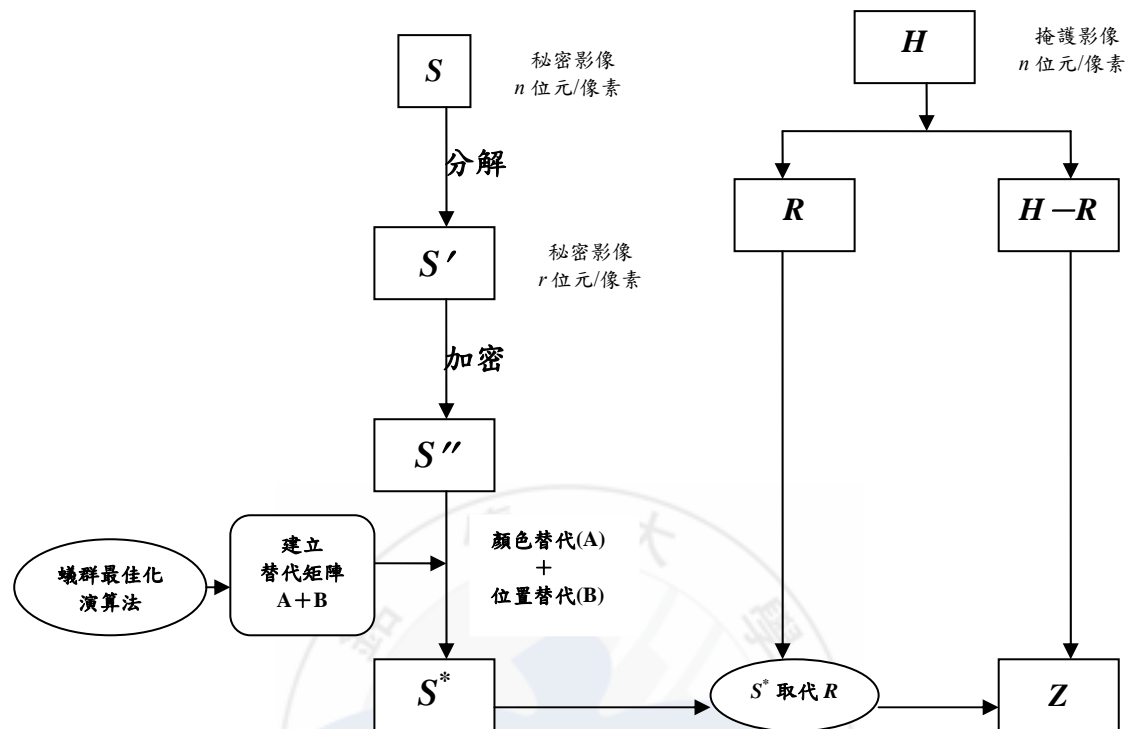


圖 16 秘密訊息隱藏流程

本論文之秘密訊息隱藏架構圖如圖 16 所示。這個架構圖主要分成左右兩大部分先各自進行，最後再合在一起變成一張藏有秘密訊息 S' 的影像 Z 。接下來我們分成左右兩個部分詳細說明。

(一)圖 16 左側的部份：

步驟一. 先將原本每個像素為 r_0 個位元的秘密訊息 S 分解成每個像素為 r 個位元的訊息 S' ，我們以圖 17(a)和(b)為例來說明，在圖 17(a)中，左邊的部分為一個 8 位元的像素，經過轉換，最後變成 4 個分別為 2 位元的像素；在圖 17(b)中，左邊的部份為 3 個 1 位元的像素，經過轉換，最後變成一個為 3 位元的像素，由這兩個例子可以知道，秘密訊息 S' 的大小會大於或小於原本的秘密訊息 S 。

$$\boxed{(00100111)_2 \xrightarrow{\text{轉換}} (00)_2 \text{和} (10)_2 \text{和} (01)_2 \text{和} (11)_2}$$

(a)

$$\boxed{(0)_2 \text{和} (1)_2 \text{和} (0)_2 \xrightarrow{\text{轉換}} (010)_2}$$

(b)

圖 17 秘密訊息轉換範例

步驟二. 透過位置重排的方式來達到加密的效果，本研究以混沌擾亂法來進行位置重排，將 S' 的座標位置打亂後成為 S'' 。混沌擾亂法的作法如公式(17)所示，其中 x 代表像素在 S' 中的原始位置， k_0 和 k_1 分別是兩個重要的參數，也相當於是密鑰的身分，因為如果有惡意竊取秘密訊息的人，在不知道 k_0 和 k_1 的情況下是無法知道原始 S' 中各像素的座標位置，而公式(16)中 p 的大小等同於秘密訊息 S 的大小。這個公式有一個限制， k_0 和 p 的最大公因數要為 1。

$$f(x) = (k_0 + k_1 \times x) \mod p \quad (16)$$

步驟三. 利用蟻群演算法去找尋顏色替代矩陣 A 及位置替代矩陣 B ，接著根據替代矩陣將 S' 轉換成 S^* 。關於蟻群演算法的求解過程，會在後面詳細描述。

(二)圖 16 右側部分

步驟一. 將掩護影像 H 分解成兩個部份，分別是每個像素為 r 位元的影像 R 和每個像素為 $(n-r)$ 位元的影像 $H-R$ 。

步驟二. 將每個像素為 r 位元的影像 R 捨棄。

步驟三. 將影像 S^* 與影像 $H-R$ 結合在一起，最後得到一張偽裝影像 Z ，這樣就

達到藏入秘密影像的目的了。

接下來我們將一一介紹最低位元置換法、替代矩陣以及如何運用蟻群最佳化演算法求解替代矩陣的詳細內容。

3.4 最低位元置換法

本論文所提出的秘密訊息隱藏方法，主要就是改良最原始的最低位元置換法，所以與最原始的最低位元置換法類似，同樣是以 S^* 來取代掩護影像 H 中每個像素的最後 r 個位元，來達到藏入秘密訊息的目的。我們利用圖 18 來說明最低位元置換法，其中 H 是一張 3×3 的影像，每一個像素有 8 個位元，而 S^* 是一張 3×3 的秘密影像，每一個像素有 2 個位元，經過最低位元置換法後， H 影像中每個像素的最後 2 個位元被 S^* 的像素所取代，最後得到下方的這一張影像 Z ，此時，下方的影像 Z 已經被藏入秘密訊息。

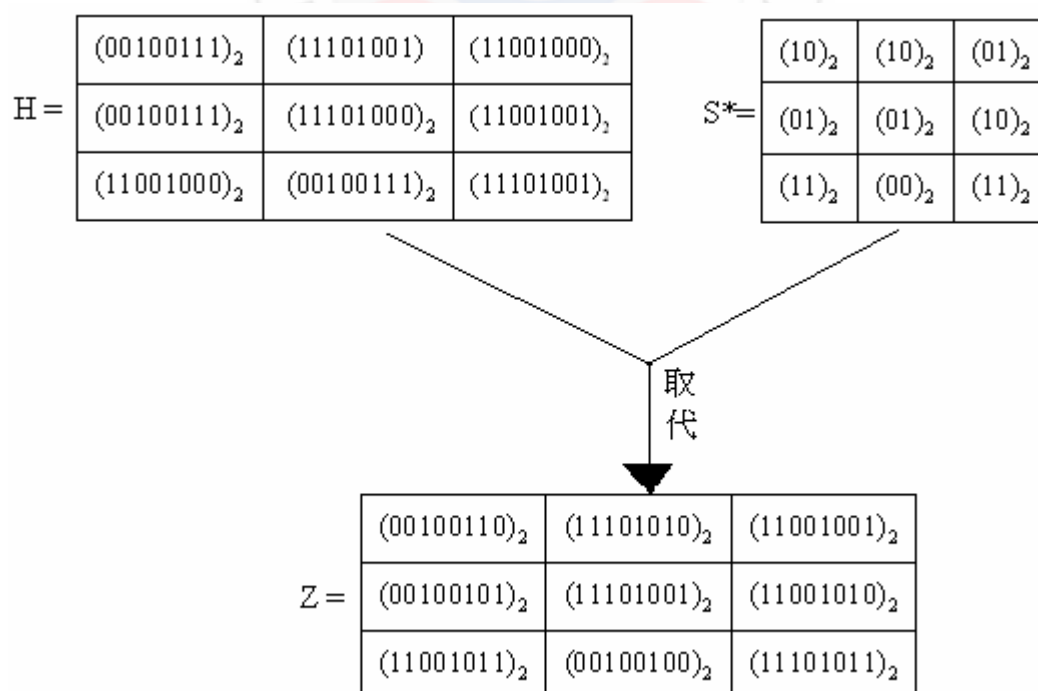


圖 18 最低位元置換法範例

3.5 替代矩陣

本論文所提到的替代矩陣是一個能夠幫助偽裝影像提昇影像品質的一個矩陣，在秘密訊息藏入掩護影像後，其影像的影像品質會變差，容易讓竊取資料者發現偽裝影像已經被藏入秘密訊息，因此為了增加安全性，可以藉由替代矩陣適當改變秘密影像的像素值及像素位置，來達到提升偽裝影像的影像品質，並增加安全性。

關於影像的品質測量方式，通常是使用信號雜訊比(Peak Signal To Noise Ratio, PSNR)來測量兩張影像的相似度，計算方式如下所示：

$$\text{PSNR} = 10 \log_{10} \frac{255^2}{\text{MSE}} \quad (17)$$

$$\text{MSE} = \left(\frac{1}{a \times b} \right) \sum_{i=0}^{a-1} \sum_{j=0}^{b-1} (I_{ij} - I'_{ij})^2 \quad (18)$$

其中 I_{ij} 表示掩護影像中座標在 (i, j) 上的像素值， I'_{ij} 表示偽裝影像中座標在 (i, j) 上的像素值， a 和 b 分別代表影像的長度和寬度。PSNR 值的數值越大，代表影像品質越好，在客觀的檢測標準上，通常只要大於或是等於 30，才可算是影像品質好，不易被惡意人士看穿，在了解如何判斷影像品質之後，接著我們介紹顏色替代矩陣與位置替代矩陣的定義。顏色替代矩陣的表示如公式(19)所示，位置替代矩陣的表示如公式(20)所示，其中 i 與 j 分別代表替代矩陣中的行與列，且 λ 的大小等於 2^r 。

$$A_{\lambda \times \lambda} = \{[a_{ij}] \mid 0 \leq i, j \leq 2^r - 1\} \quad (19)$$

$$B_{\mu \times \mu} = \{[b_{ij}] \mid 0 \leq i, j \leq d\} \quad (20)$$

其中 $a_{ij} \in \{0, 1\}$ 且 $b_{ij} \in \{0, 1\}$ ，替代矩陣是一個只有 0 和 1 所組成的一個矩陣，而且在替代矩陣 A 及 B 中，每一行只能有一個 1，且每一列也只能有一個 1，其餘全部為 0。如圖 19(a)的矩陣中，第一列也有兩個 1，第三列卻沒有任何一個 1，所以圖 19(a)的矩陣不是一個合格的替代矩陣；至於圖 19(b)的矩陣，因為每一行只有一個 1，且每一列也只有一個 1，因此，它是一個合格的替代矩陣。

$$\begin{matrix} \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} & \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\ \text{(a)} & \text{(b)} \end{matrix}$$

圖 19 (a) 不合格的替代矩陣；(b) 合格的替代矩陣

而顏色替代矩陣中的 1 和 0 所代表的意義指的是：如果矩陣中的元素 a_{ij} 為 1，代表影像 S'' 中的像素值 i 值會被 j 值所取代，若 a_{ij} 為 0 則代表不作任何變動。因此，如果顏色替代矩陣是圖 19(b)，且 S'' 的大小是 2×2 ，表示如圖 20 左半部，根據圖 19(b)的座標中，有 1 的座標分別為 (0,0)，(1,2)，(2,1)，(3,3)，所以 S'' 中的像素值，0 以 0 取代，1 以 2 取代，2 以 1 取代，3 以 3 取代，最後可以得到一個新的秘密影像 S^* ，由圖 20 右半部所表示。

$$S'' = \begin{bmatrix} 1 & 3 \\ 0 & 2 \end{bmatrix}_{10} \xrightarrow{\text{轉換}} S^* = \begin{bmatrix} 2 & 3 \\ 0 & 1 \end{bmatrix}_{10}$$

圖 20 顏色替代矩陣轉換範例

而位置替代矩陣中的 1 和 0 所代表的意義指的是：如果矩陣中的元素 b_{ij} 為 1，代表影像 S'' 中的大小相同的區塊 i 會被區塊 j 所取代，若 b_{ij} 為 0 則代表不作任何變動。如果顏色替代矩陣是圖 19(b)，且 S'' 的大小是 2×2 ，先分割成 2×2 大小相同的區塊，剛好每一個區塊只有一個像素，表示如圖 22 左半部，根據圖 19(b) 的座標中，有 1 的座標分別為 $(0,0)$ ， $(1,2)$ ， $(2,1)$ ， $(3,3)$ ，所以 S'' 中的區塊位置，0 以 0 取代，1 以 2 取代，2 以 1 取代，3 以 3 取代，最後可以得到一個新的秘密影像 S^* ，由圖 21 右半部所表示。

$$S'' = \begin{bmatrix} 1 & 3 \\ 0 & 2 \end{bmatrix}_{10} \xrightarrow{\text{轉換}} S^* = \begin{bmatrix} 1 & 0 \\ 3 & 2 \end{bmatrix}_{10}$$

圖 21 位置替代矩陣轉換範例

3.6 結合像素與位置替代矩陣及其討論

由於只應用顏色替代矩陣來改善偽裝影像的影像品質是不足夠的，因此本論文又再提出了結合像素與位置替代矩陣共同來改善偽裝影像的影像品質。其中，結合像素與位置替代矩陣並非只是單獨的分開的求解顏色替代矩陣和位置替代矩陣，而是求解具有相互關係的顏色替代矩陣與位置替代矩陣，接著介紹詳細的內容。

如圖 22 所示，其中圖 22(A)是一張 2×2 的掩護影像，其像素值分別是 $(1,1,2,3)$ ；圖 22(B)是一張 2×2 的秘密訊息，經由最佳的顏色替代矩陣轉換之後再經過最佳位置替代矩陣轉換後所得到的結果；圖 22(C)是一張 2×2 的秘密訊息，經由具有相互關係的像素與位置替代矩陣轉換之後所得到的結果。由圖 22(B)中可以得知，秘密訊息 $(3,1,0,0)$ 經過顏色替代矩陣轉換後變成 $(3,1,2,2)$ ，其中改變了 2 個像素值，使得秘密訊息原本只有 1 個像素值與掩護影像相同改善成有 2

個相同的像素值，接著再經過位置替代矩陣的轉換使秘密訊息轉換成(2,1,2,3)，與掩護影像比較後發現已有 3 個像素值是相同的，由此可知，運用具有相互關係的顏色替代矩陣與位置替代矩陣與只運用顏色替代矩陣相比較，前者更可以改善偽裝影像被破壞的程度。

雖然加入位置替代矩陣可以改善偽裝影像的影像品質，但是轉變後的秘密訊息的像素值組合還是與掩護影像的像素值組合有所差異，因此本論文為了能更加改善偽裝影像的影像品質，而提出了結合顏色替代矩陣與位置替代矩陣來共同改善偽裝影像的影像品質，由圖 22(C)中可以知道其顏色替代矩陣與位置替代矩陣

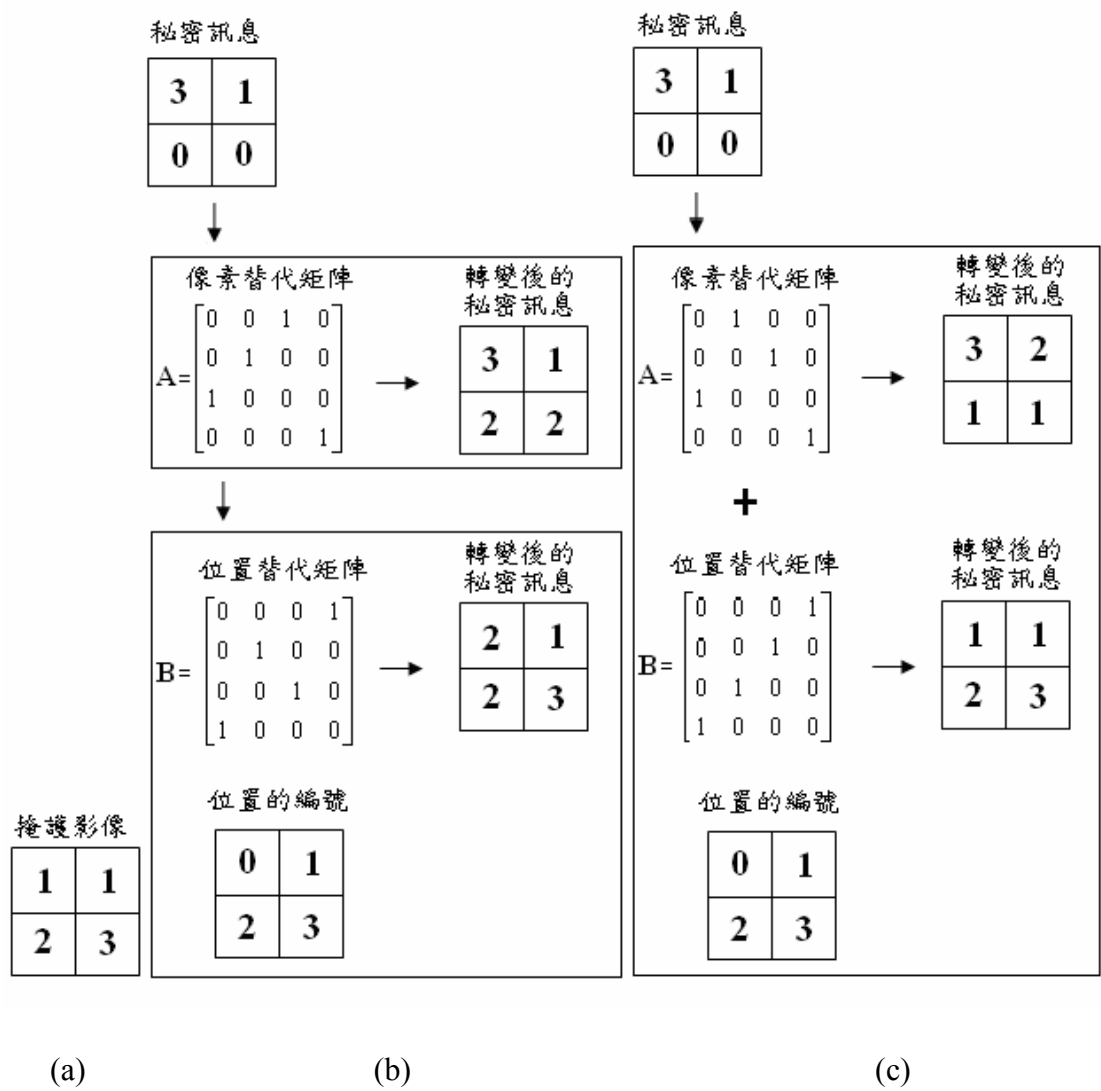


圖 22 顏色替代矩陣與位置替代矩陣關係之範例

是有相互關係的，而非單獨去求解其最佳的替代矩陣，是具有相輔相成的關係，經過轉換後秘密訊息(3,1,0,0)變成(1,1,2,3)與掩護影像是完全相同的，也就是說掩護影像在藏入秘密訊息後卻完全不會破壞其影像的品質。由此可知，本論文所提出的想法對資訊隱藏有很大的幫助。

3.7 運用蟻群演算法找尋替代矩陣

本論文尋找替代矩陣的方法是使用蟻群最佳化演算法來建立替代矩陣。接下來我們將敘述替代矩陣如何與蟻群最佳化演算法產生連結；此外，我們也將介紹蟻群最佳化演算法的詳細計算過程。

3.7.1 蟻群最佳化演算法與替代矩陣之間的關聯

在這一小節中，我們將介紹蟻群最佳化演算法如何與替代矩陣產生關聯。蟻群最佳化演算法的第一步是建立一個連通的無向圖(Undirected Connected-graph)，而這個圖就等於我們所要求解的問題圖。在圖中有許許多多的頂點(vertices)，而螞蟻所需要做的行為就是在這許許多多的頂點中建立它們的旅程，也就是說，假設螞蟻正要出巢穴去找尋食物，此時，它必須決定下一個要去的頂點為那一個頂點，走完這個頂點後仍然還要決定下一個頂點即將前往的為何，直到螞蟻走到最終食物點為止。以 4×4 的替代矩陣為例，蟻群演算法所建立的圖形如圖 23 所示，其中圖形中的頂點 V_{ij} 對應於替代矩陣的元素 a_{ij} 。螞蟻從巢穴出發會面臨到四個頂點 V_{00} ， V_{10} ， V_{20} 和 V_{30} ，所以螞蟻現在將有四種選擇，如果螞蟻的選擇是 V_{00} ，則第二步他將面臨到三種選擇，分別是 V_{11} ， V_{21} 和 V_{31} ，如果螞蟻的選擇是 V_{11} ，則接著第三步他所面臨的選擇是 V_{02} ， V_{22} 和 V_{32} ，如果螞蟻的選擇是 V_{02} ，則第四步他所面臨的選擇是 V_{13} ， V_{23} 和 V_{33} ，最後到達食物區。依此類推，螞蟻最初的選擇都有 V_{00} ， V_{10} ， V_{20} 和 V_{30} 這四種，不論選擇哪一種，到了第二步也還是有三種選擇，第三步和第四步也同樣是三種選擇，而最後一步

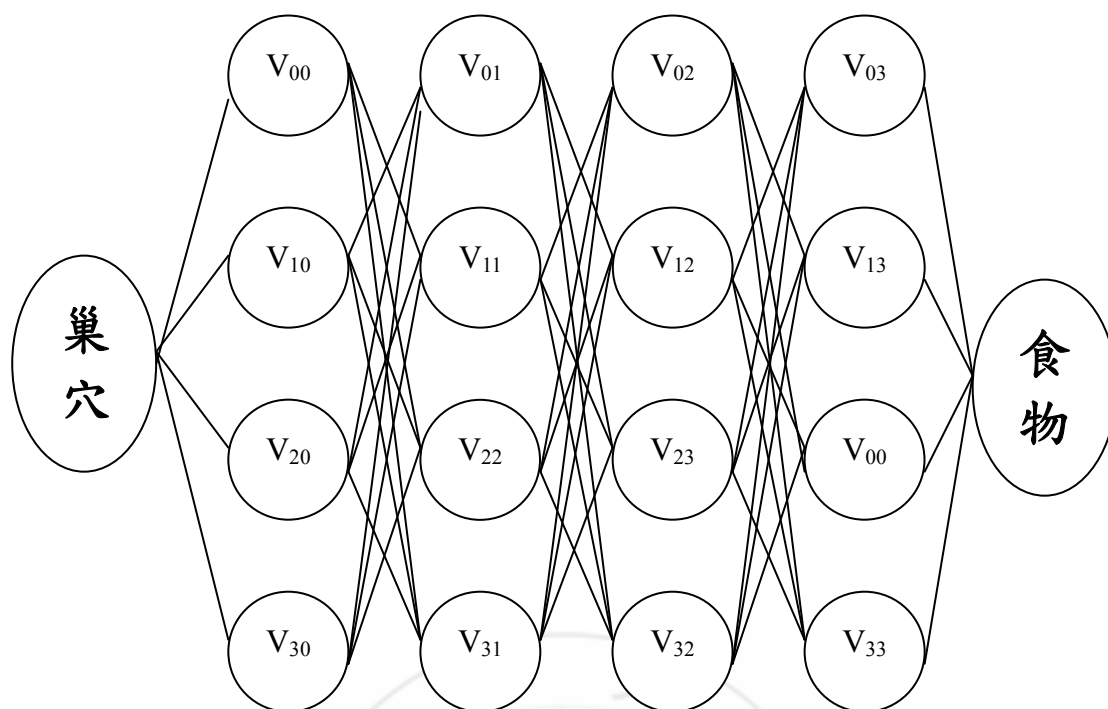


圖 23 4×4 替代矩陣的蟻群最佳化演算法問題圖

就是到達食物區一種選擇。

在上述的例子中，如果螞蟻選擇頂點 V_{ij} ，對應到替代矩陣上，也就是相當於替代矩陣中元素 a_{ij} 為 1 的意思，而其餘螞蟻沒有選擇的頂點則代表替代矩陣中元素為 0 的意思，因此上述例子相當於求解一個 4×4 的替代矩陣。

但是在上述例子中，出現了一個問題，那就是可能會產生不合格的矩陣，也就是違反替代矩陣中每一行只能有一個 1 且每一列也只能有一個 1 的限制條件，因此，在下一小節中，我們將利用蟻群演算法中的費洛蒙更新來避免違反替代矩陣的限制條件。

3.7.2 以蟻群演算法求解替代矩陣

經由上一小節的描述，我們可以知道利用蟻群演算法可以求解替代矩陣，因此，這一小節我們將詳細敘述蟻群演算法的求解過程。

步驟一：設定目標函數，替代矩陣的最終目標就是提高偽裝影像的影像品質，因此所要設定的目標就是盡可能使 PSNR 值達到最大，或是盡可能使 MSE 達到最小，所以目標如公式(21)所示。

$$\text{Min } f(I') = \sum_{i=0}^{a-1} \sum_{j=0}^{b-1} (I_{ij} - I'_{ij})^2 \quad (21)$$

步驟二：建立替代矩陣的問題圖。

步驟三：定義參數。

τ_0 ：費洛蒙初始值，且費洛蒙初始值為零。

τ_{ij} ：點 i 到點 j 之間的費洛蒙值。

τ^{best} ：是目前最佳路徑上的費洛蒙值。

T ：執行蟻群演算法的次數。

m ：執行蟻群最佳化演算法的總螞蟻數。

η_{ij} ：是點 i 到點 j 的啟發值。

p_{ij}^k ：是第 k 隻螞蟻在點 i 要選擇下一個點 j 的機率。

N_i^k ：是第 k 隻螞蟻在點 i 上所面臨的鄰近頂點。

q_0 ：一個隨機參數，如果隨機值小於 q_0 ，則按照公式(22)來選取路線，反之，螞蟻則隨機行走。

ρ ：是費洛蒙蒸發參數，如果值越大，則費洛蒙蒸發速率越快，相反的，如果值越小，則費洛蒙蒸發速率越慢，而且 ρ 介於 0 到 1 之間。

α ：調整參數之一，如果 α 值越大，則代表在選擇下一個點的機率中，費洛蒙的濃度值越重要，相反的，如果 α 值越小，則代表費洛蒙濃

度值越不重要。

β ：調整參數之一，如果 β 值越大，則代表在選擇下一個點的機率中，

啟發值越重要，相反的，如果 β 值越小，則代表啟發值越不重要。

步驟四：建立旅程，蟻群可以利用公式(22)所定義的機率公式來判斷下一個要行走的點是哪一個點，計算出來的機率越大，代表下一個可能會選擇的機率越大，且在公式(22)當中描述到，如果某頂點不在 N_i^k 中，那麼行走到某頂點的機率就為零。

$$p_{ij}^k = \frac{[\tau_{ij}]^\alpha [\eta_{ij}]^\beta}{\sum_{l \in N_i^k} [\tau_{il}]^\alpha [\eta_{il}]^\beta}, \text{ if } j \in N_i^k. \quad (22)$$

步驟五：區域更新費洛蒙，其更新規則如公式(23)，執行費洛蒙局部更新可以避免螞蟻一直行走在同一路線上，或是可以避免掉行走不必要的迴圈。

$$\tau_{ij}(t+1) = (1-\rho) \times \tau_{ij}(t) + \rho \times \tau_0 \quad (23)$$

步驟六：全域更新費洛蒙，其更新的規則如公式(24)，執行費洛蒙全域更新可以幫助螞蟻不會花費太多的時間在其他較差的路徑上。並且可以在這些較好的路徑上找到更好的路徑。

$$\tau^{best}(t+1) = (1-\rho) \times \tau^{best}(t) + \rho \left(\frac{1}{\min \text{MSE}} \right) \quad (24)$$

步驟七：重複執行步驟四到步驟六，直到重複的次數到達實驗所設定的次數為止

步驟八：將求得替代矩陣應用在 S'' 的轉換上。

3.8 秘密訊息的取出

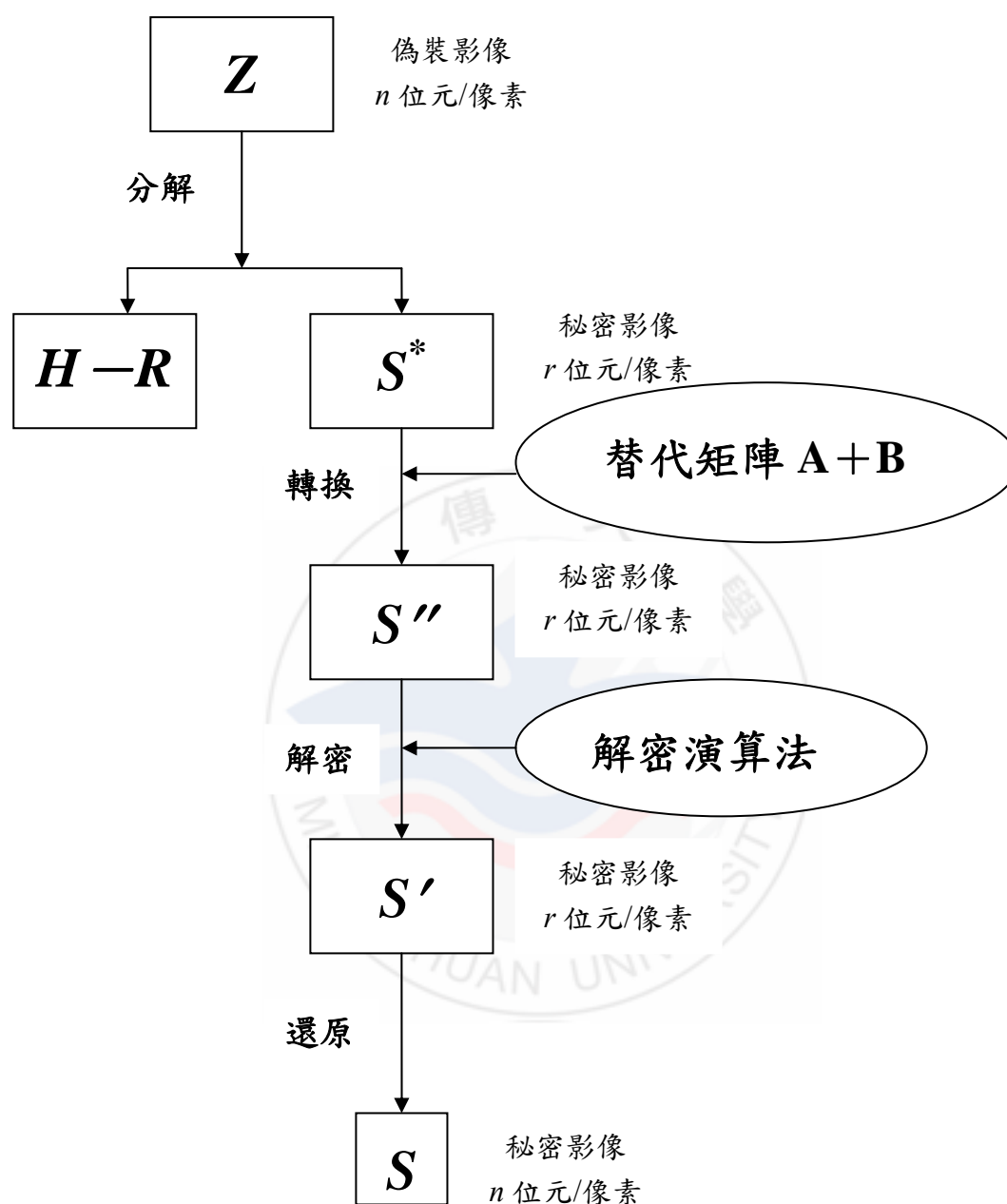


圖 24 秘密訊息偵測流程

這一小節我們介紹如何將秘密訊息取出，取出秘密訊息的流程如圖 24 所示，接下來我們將一步一步的詳細敘述取出過程。

步驟一：將藏有秘密訊息且每個像素有 n 個位元的偽裝影像 Z 分解成每個像

素有 $n-r$ 位元的 $Z-S^*$ 影像和每個像素有 r 個位元的秘密訊息 S^* 。

步驟二：將秘密訊息 S^* 利用替代矩陣轉換秘密訊息 S^* 中部份像素的像素值及位置。轉變後成為秘密訊息 S'' 。

步驟三：將秘密訊息 S'' 利用混沌擾亂法還有所設定的參數 k_0 和 k_1 可以得到 $f(x)$ 的值，再利用得到的值與公式(25)及(26)將 S'' 中各像素值目前所在位置轉換，轉換後秘密訊息 S'' 變成秘密訊息 S' 。

$$X = (f(x) + 1) \bmod n \quad (25)$$

$$Y = \lfloor (f(x) + 1) / m \rfloor \quad (26)$$

步驟四：將每個像素值為 r 位元的秘密訊息 S' 重新組合成每個像素值為 r_0 位元的秘密訊息 S ，而 S 就是我們所要得到的最原始的秘密訊息。

第肆章實驗結果與分析

4.1 實驗簡介

本論文使用 Borland C++ Builder 來實作未使用替代矩陣與有應用蟻群最佳化演算法來產生替代矩陣以藏入秘密訊息的實驗。系統配備為個人電腦 CPU 2.8 GHz，記憶體為 2.5GB，作業系統為 Windows XP。

本論文實驗共分成兩大部分，第一部分是每張掩護影像的每個像素皆藏入 2 位元的秘密訊息，實驗的參數設定如下：螞蟻數量 $m = 3$ 、螞蟻往返的總次數 $T = 300$ 次、費洛蒙初始值 $\tau_0 = 0.5$ 、 $\alpha = 1$ 、 $\beta = 1$ 、 $\rho = 0.8$ 、 $q_0 = 0.3$ ；第二部份是每張掩護影像的每個像素皆藏入 4 位元的秘密訊息，實驗的參數設定如下：螞蟻數量 $m = 8$ 、螞蟻往返的總次數 $T = 400$ 次、費洛蒙初始值 $\tau_0 = 0.5$ 、 $\alpha = 1$ 、 $\beta = 1$ 、 $\rho = 0.8$ 、 $q_0 = 0.3$ ，其實驗結果將分別於 4.2 節與 4.3 節呈現。

在本論文中，我們使用 PSNR 值的高低來判斷影像品質的好壞。

4.2 掩護影像的每個像素藏入 2 位元秘密訊息的實驗結果

在本節中我們將呈現每張掩護影像的每個像素皆藏入 2 位元的秘密訊息的實驗結果。我們總共分成 4 種不同的實驗，第一種實驗是未使用任何的替代矩陣來藏入秘密訊息；第二種實驗是只使用 4×4 的顏色替代矩陣來轉變秘密訊息；第三種實驗是使用具有相互關係的 4×4 的顏色替代矩陣與 4×4 的位置替代矩陣來轉變秘密訊息；第四種實驗是使用具有相互關係的 4×4 的顏色替代矩陣與 16×16 的位置替代矩陣來轉變秘密訊息。

在 4.2 的實驗中，我們分別使用 17 張不同的掩護影像，大小為 512×512 來藏入 1 種秘密訊息，大小為 256×256 ，實驗結果如下列各表所示：

1. Airplane

表 1 Airplane 影像介紹 (藏 2 位元)

影像類別	掩護影像	秘密訊息
影像名稱	Airplane	秘密訊息 1
影像大小	512×512	256×256
影像		

表 2 Airplane 實驗數據(藏 2 位元)

方法	PSNR 值
無替代矩陣	44.375576
4×4 顏色替代矩陣	44.382295
4×4 顏色替代矩陣與 4×4 位置替代矩陣	44.393490
4×4 顏色替代矩陣與 16×16 位置替代矩陣	44.411240

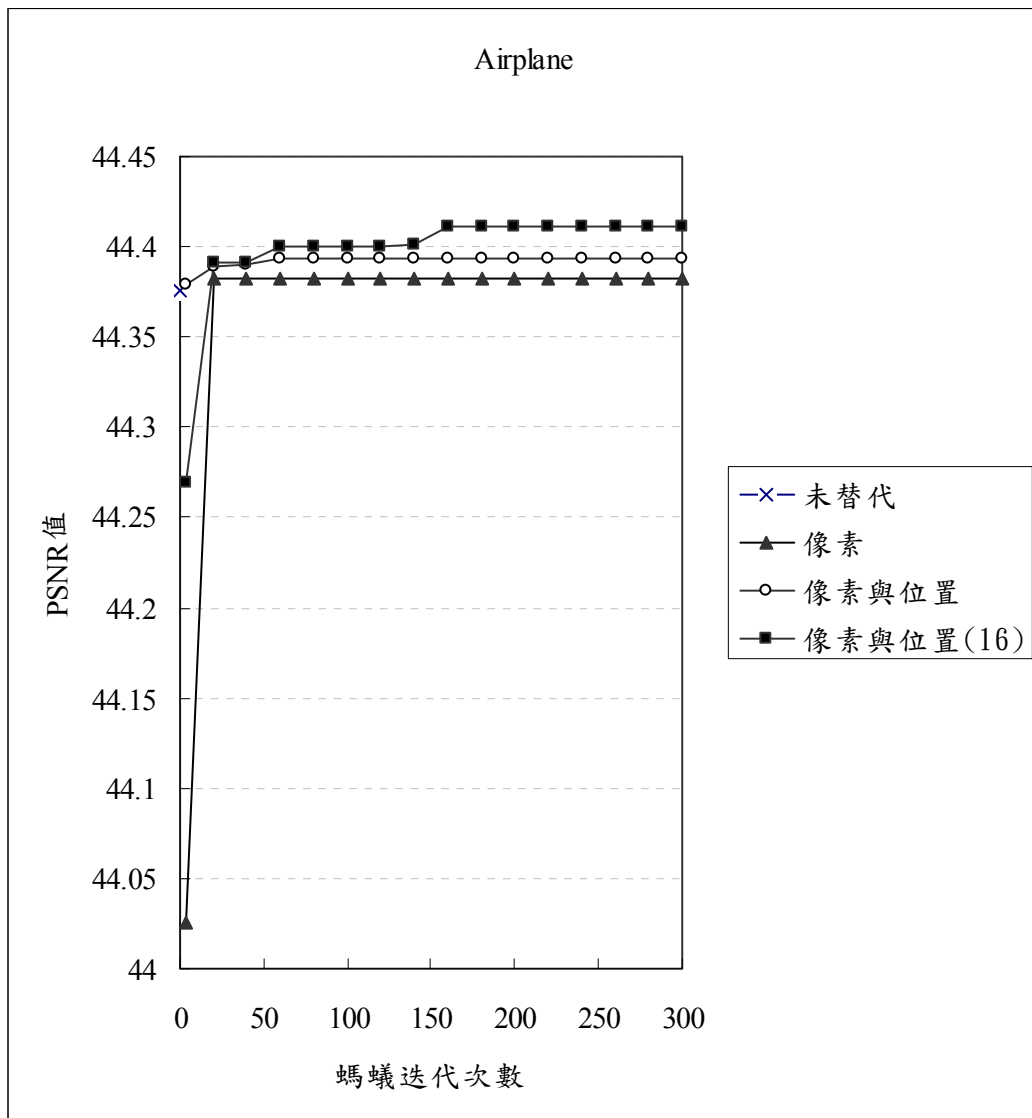


圖 25 Airplane 螞蟻迭代次數與 PSNR 值分佈圖 (藏入 2 位元)

2. 掩護影像：Baboon

表 3 Baboon 影像介紹 (藏 2 位元)



影像類別	掩護影像	秘密訊息
影像名稱	Baboon	秘密訊息 1
影像大小	512×512	256×256
影像		

表 4 Baboon 實驗數據 (藏 2 位元)

方法	PSNR 值
無替代矩陣	44.388840
4×4 顏色替代矩陣	44.388840
4×4 顏色替代矩陣與 4×4 位置替代矩陣	44.404785
4×4 顏色替代矩陣與 16×16 位置替代矩陣	44.414973

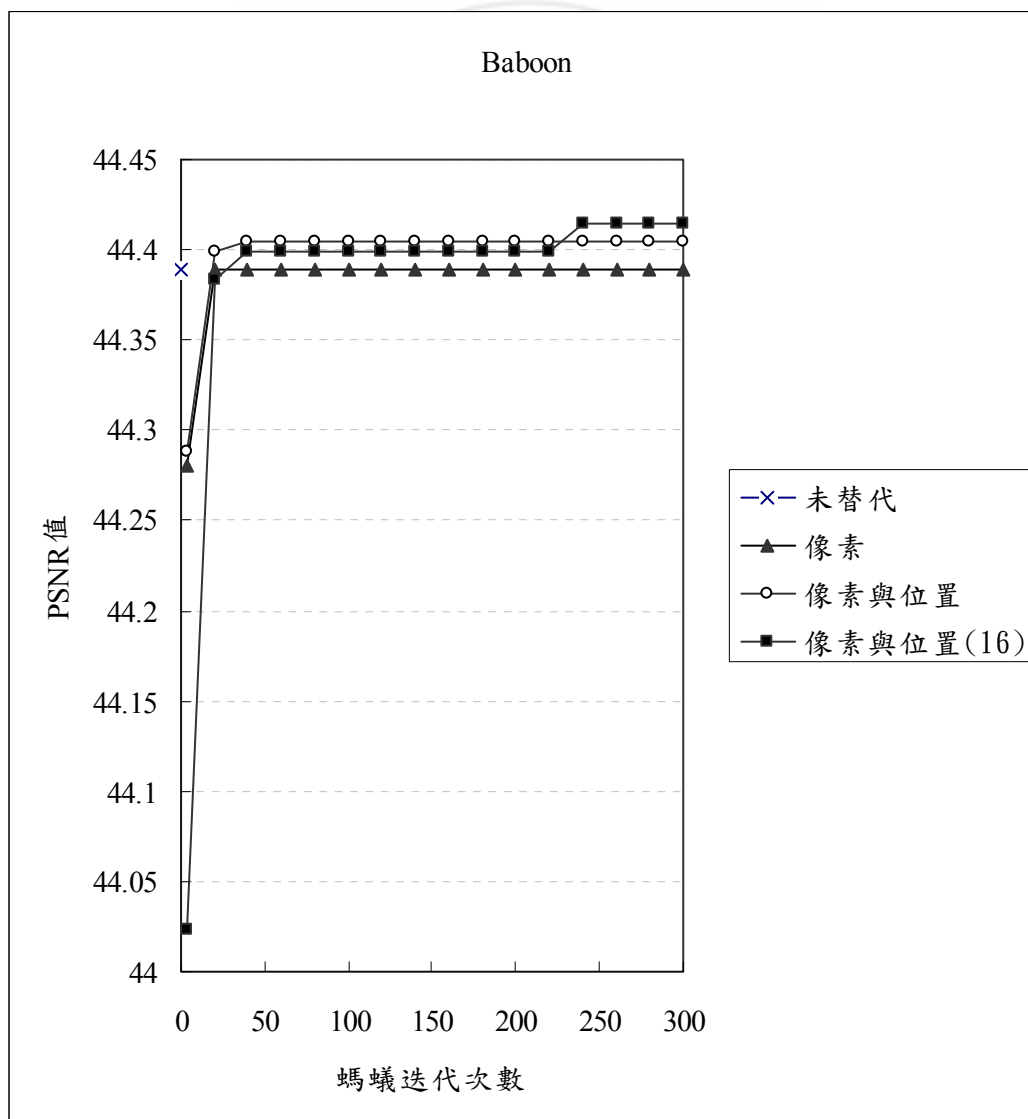


圖 26 Baboon 螞蟻迭代次數與 PSNR 值分佈圖 (藏入 2 位元)

3. 掩護影像： Bird

表 5 Bird 影像介紹（藏 2 位元）

影像類別	掩護影像	秘密訊息
影像名稱	Bird	秘密訊息 1
影像大小	512×512	256×256
影像		

表 6 Bird 實驗數據（藏 2 位元）

方法	PSNR 值
無替代矩陣	44.392117
4×4 顏色替代矩陣	44.398034
4×4 顏色替代矩陣與 4×4 位置替代矩陣	44.412521
4×4 顏色替代矩陣與 16×16 位置替代矩陣	44.438229

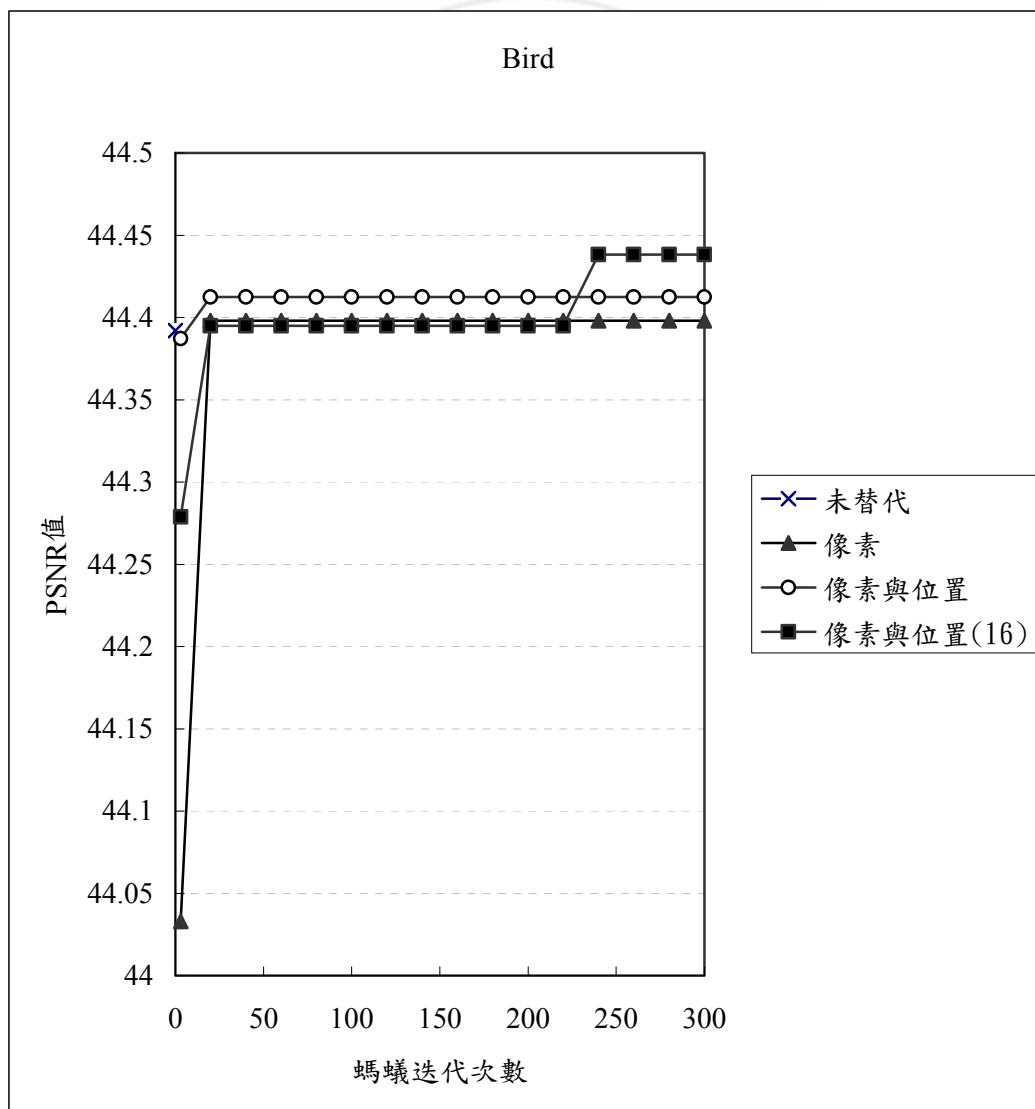


圖 27 Bird 螞蟻迭代次數與 PSNR 值分佈圖（藏入 2 位元）

4. 掩護影像：Bird1

表 7 Bird1 影像介紹（藏 2 位元）

影像類別	掩護影像	秘密訊息
影像名稱	Bird1	秘密訊息 1
影像大小	512×512	256×256
影像		

表 8 Bird1 實驗數據（藏 2 位元）

方法	PSNR 值
無替代矩陣	44.324990
4×4 顏色替代矩陣	44.324990
4×4 顏色替代矩陣與 4×4 位置替代矩陣	44.337421
4×4 顏色替代矩陣與 16×16 位置替代矩陣	44.347672

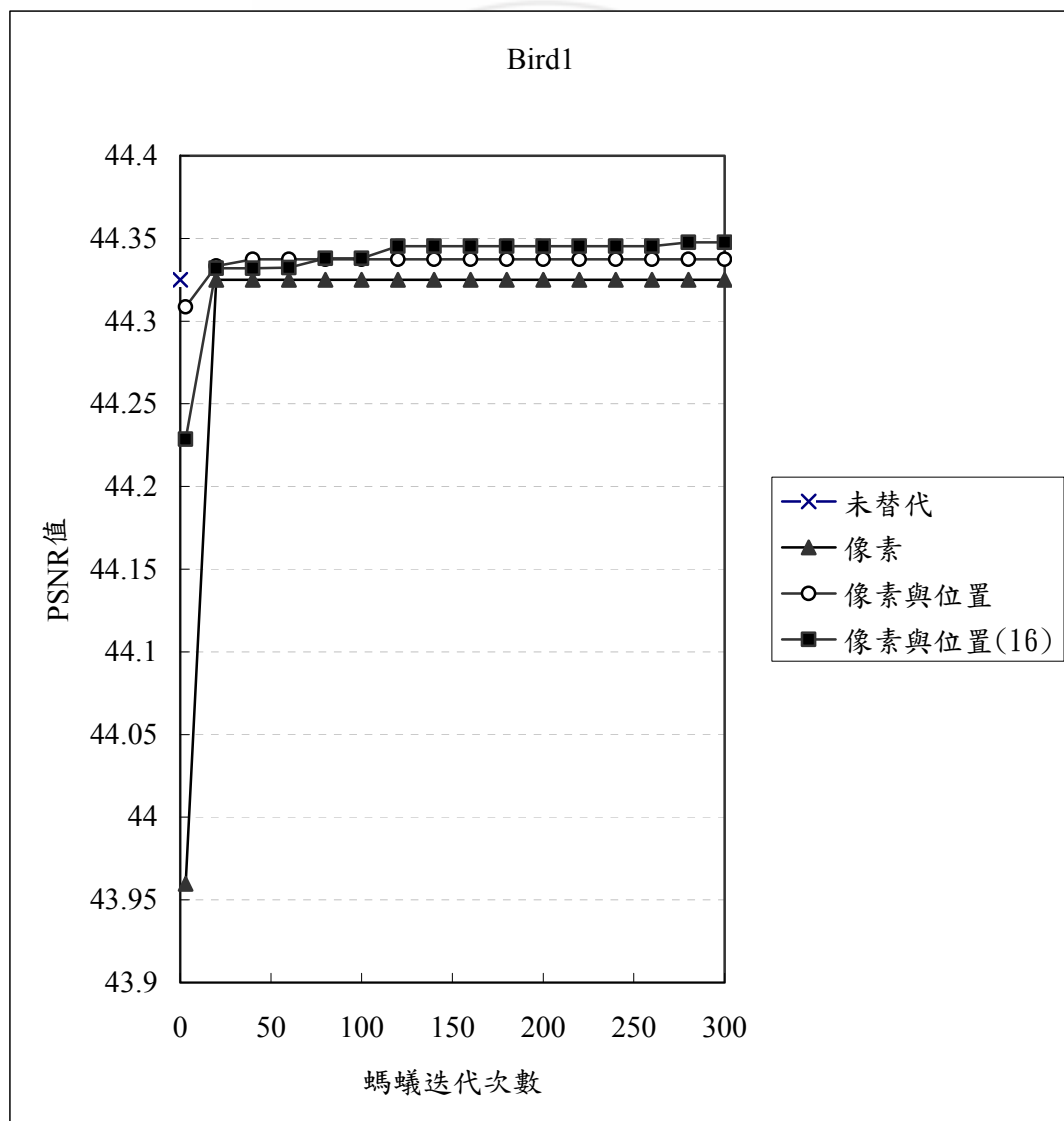


圖 28 Bird1 螞蟻迭代次數與 PSNR 值分佈圖（藏入 2 位元）

5. 掩護影像：Boat

表 9 Boat 影像介紹（藏 2 位元）

影像類別	掩護影像	秘密訊息
影像名稱	Boat	秘密訊息 1
影像大小	512×512	256×256
影像		

表 10 Boat 實驗數據（藏 2 位元）

方法	PSNR 值
無替代矩陣	44.351784
4×4 顏色替代矩陣	44.356409
4×4 顏色替代矩陣與 4×4 位置替代矩陣	44.374302
4×4 顏色替代矩陣與 16×16 位置替代矩陣	44.379589

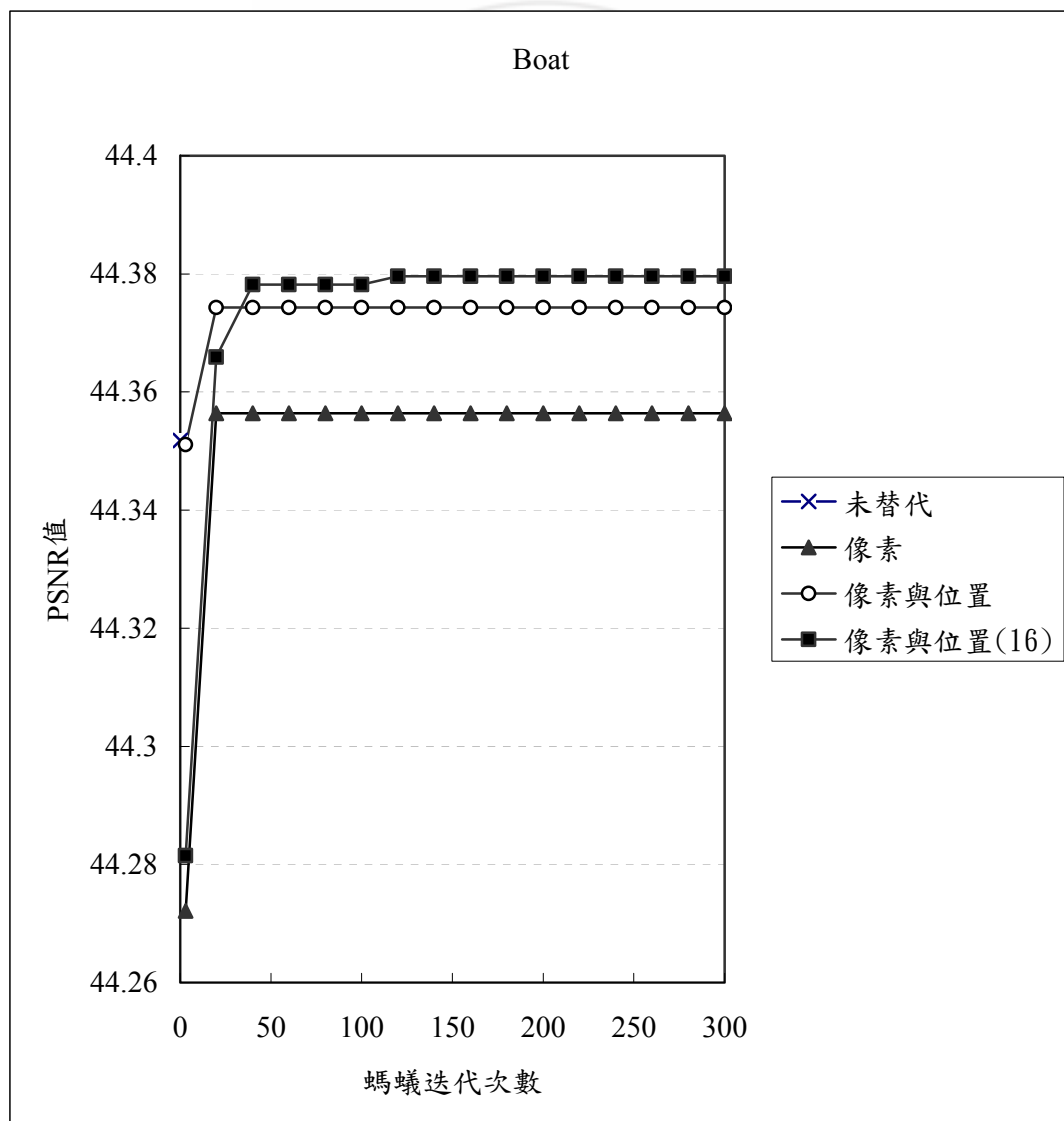


圖 29 Boat 螞蟻迭代次數與 PSNR 值分佈圖（藏入 2 位元）

6. 掩護影像：Cat

表 11 Cat 影像介紹（藏 2 位元）

影像類別	掩護影像	秘密訊息
影像名稱	Cat	秘密訊息 1
影像大小	512×512	256×256
影像		

表 12 Cat 實驗數據（藏 2 位元）

方法	PSNR 值
無替代矩陣	44.966405
4×4 顏色替代矩陣	44.966405
4×4 顏色替代矩陣與 4×4 位置替代矩陣	44.983356
4×4 顏色替代矩陣與 16×16 位置替代矩陣	44.994637

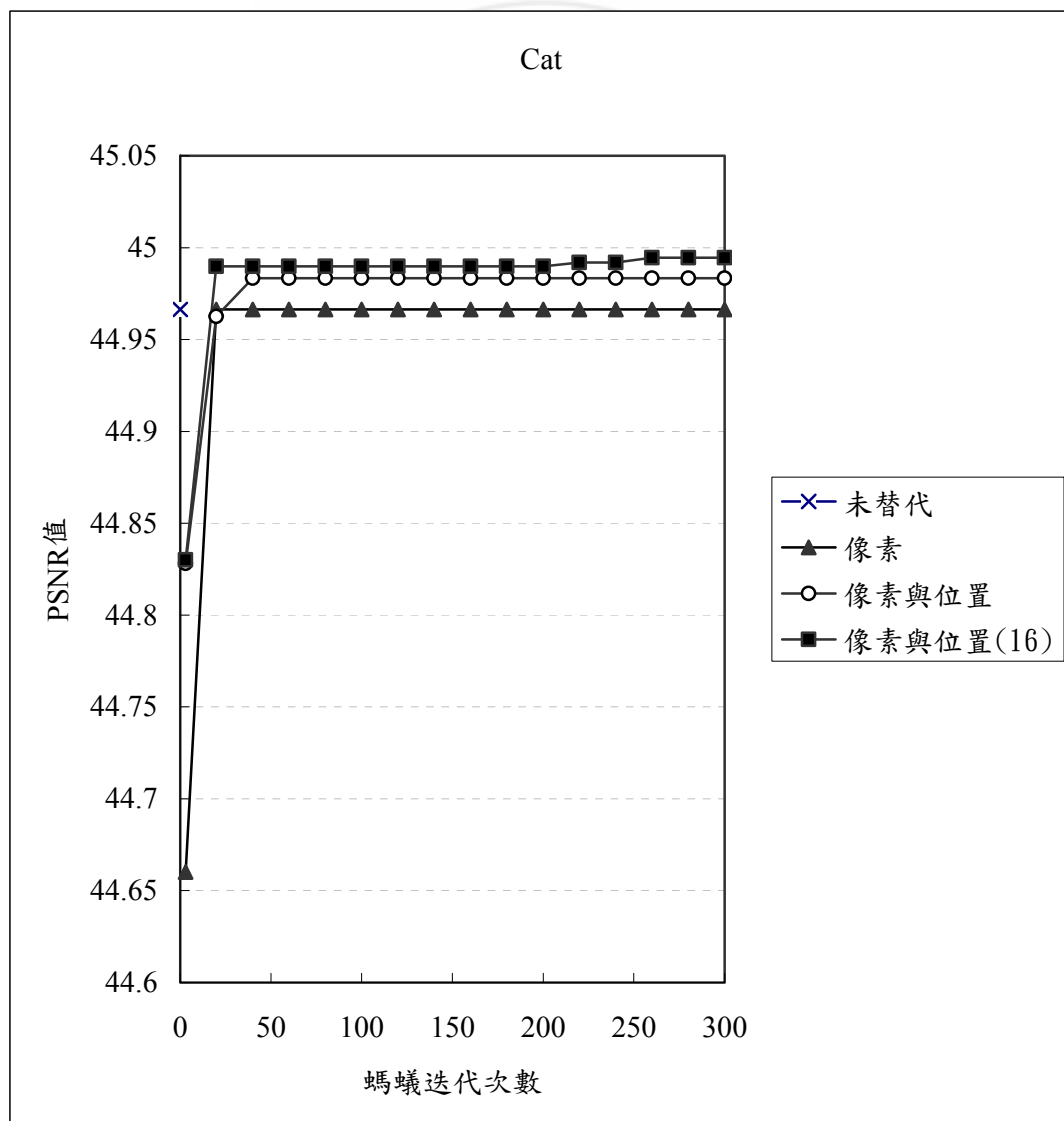


圖 30 Cat 螞蟻迭代次數與 PSNR 值分佈圖（藏入 2 位元）

7. 掩護影像：Girl

表 13 Girl 影像介紹（藏 2 位元）

影像類別	掩護影像	秘密訊息
影像名稱	Girl	秘密訊息 1
影像大小	512×512	256×256
影像		

表 14 Girl 實驗數據（藏 2 位元）

方法	PSNR 值
無替代矩陣	44.381283
4×4 顏色替代矩陣	44.389632
4×4 顏色替代矩陣與 4×4 位置替代矩陣	44.401043
4×4 顏色替代矩陣與 16×16 位置替代矩陣	44.405720

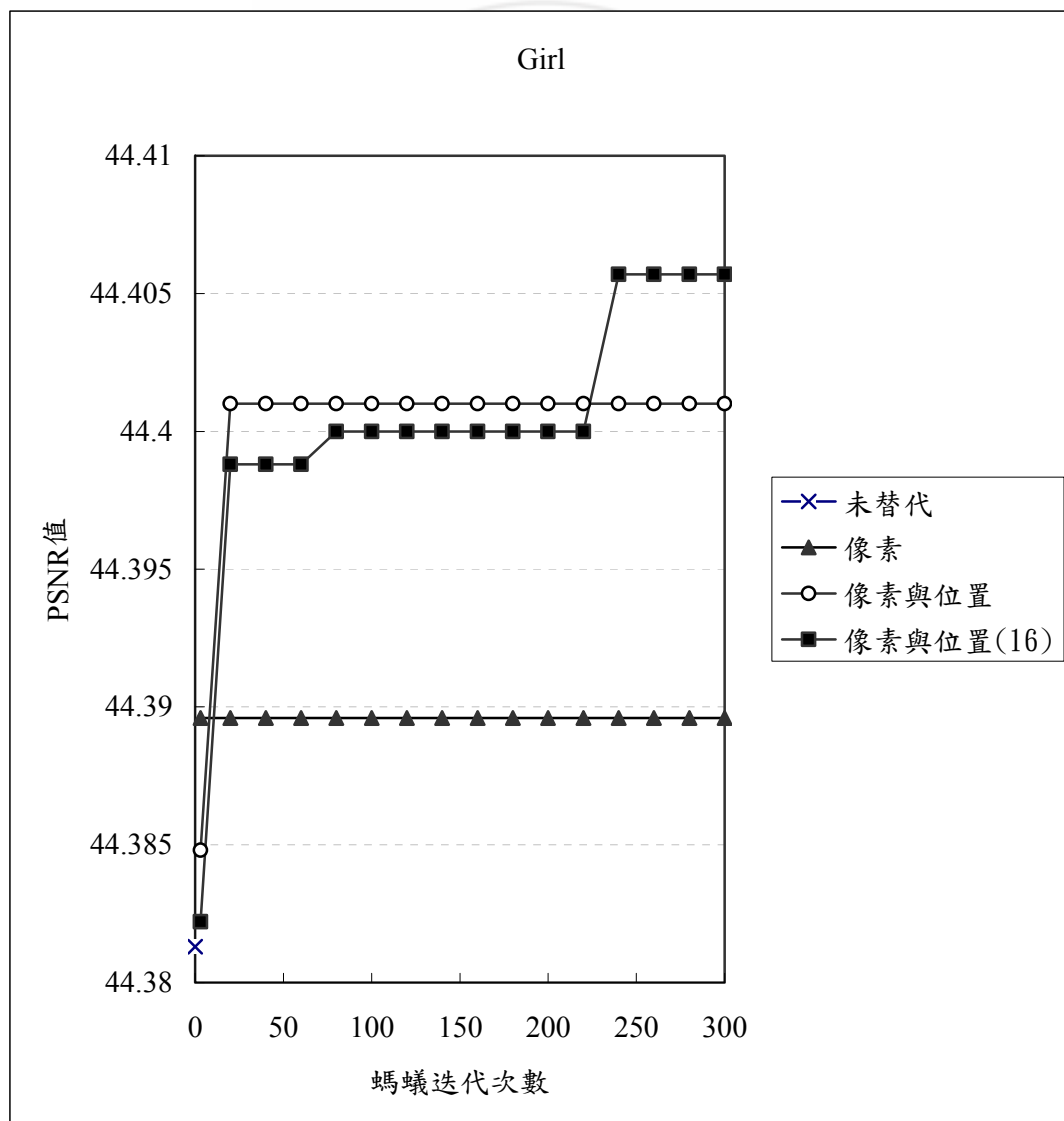


圖 31 Girl 螞蟻迭代次數與 PSNR 值分佈圖（藏入 2 位元）

8. 掩護影像：Lena

表 15 Lena 影像介紹（藏 2 位元）

影像類別	掩護影像	秘密訊息
影像名稱	Lena	秘密訊息 1
影像大小	512×512	256×256
影像		

表 16 Lena 實驗數據（藏 2 位元）

方法	PSNR 值
無替代矩陣	44.382568
4×4 顏色替代矩陣	44.389107
4×4 顏色替代矩陣與 4×4 位置替代矩陣	44.397923
4×4 顏色替代矩陣與 16×16 位置替代矩陣	44.417633

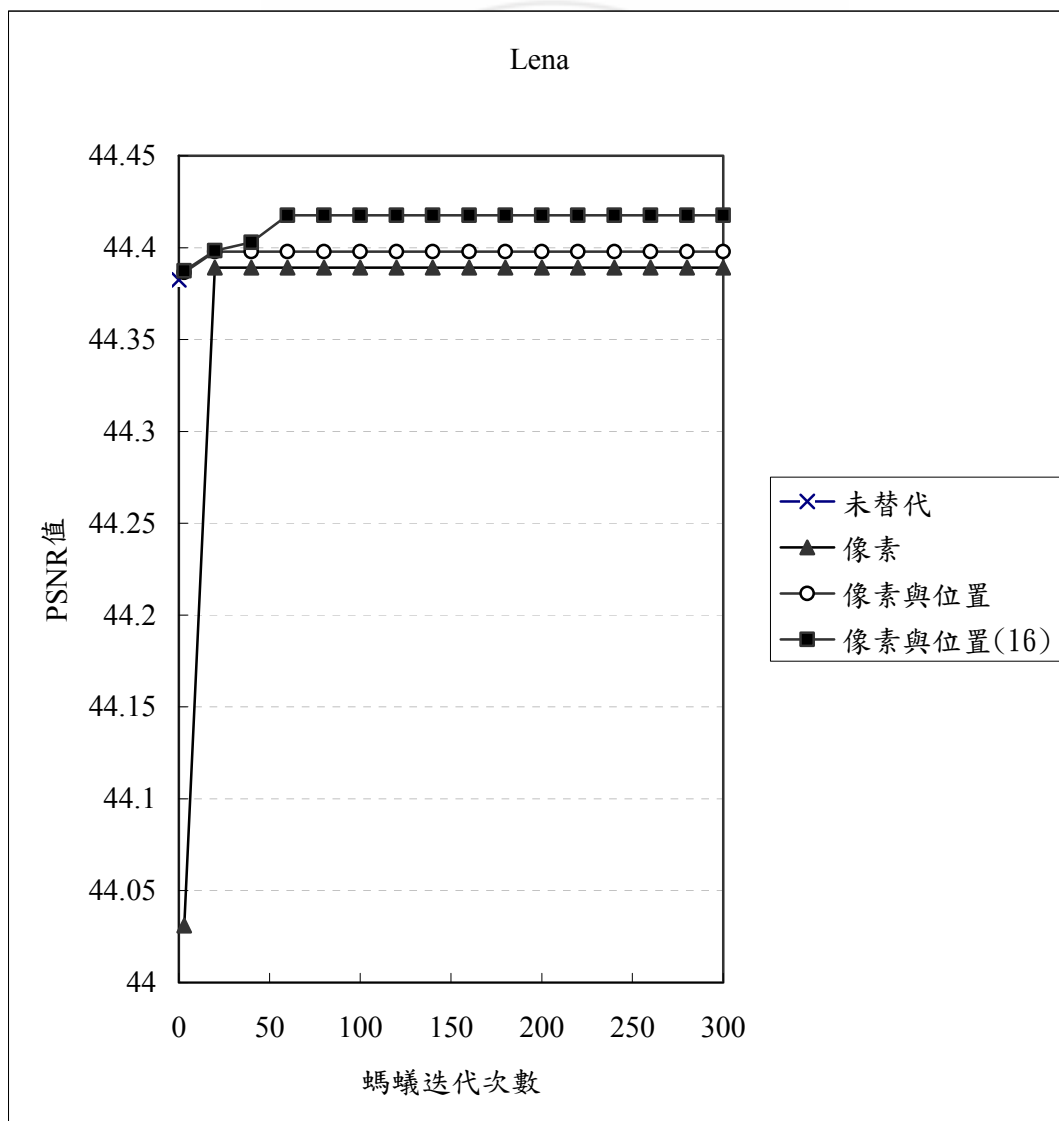


圖 32 Lena 螞蟻迭代次數與 PSNR 值分佈圖（藏入 2 位元）

9. 掩護影像：Lenna

表 17 Lenna 影像介紹（藏 2 位元）

影像類別	掩護影像	秘密訊息
影像名稱	Lenna	秘密訊息 1
影像大小	512×512	256×256
影像		

表 18 Lenna 實驗數據（藏 2 位元）

方法	PSNR 值
無替代矩陣	44.256804
4×4 顏色替代矩陣	44.385525
4×4 顏色替代矩陣與 4×4 位置替代矩陣	44.385525
4×4 顏色替代矩陣與 16×16 位置替代矩陣	44.394836

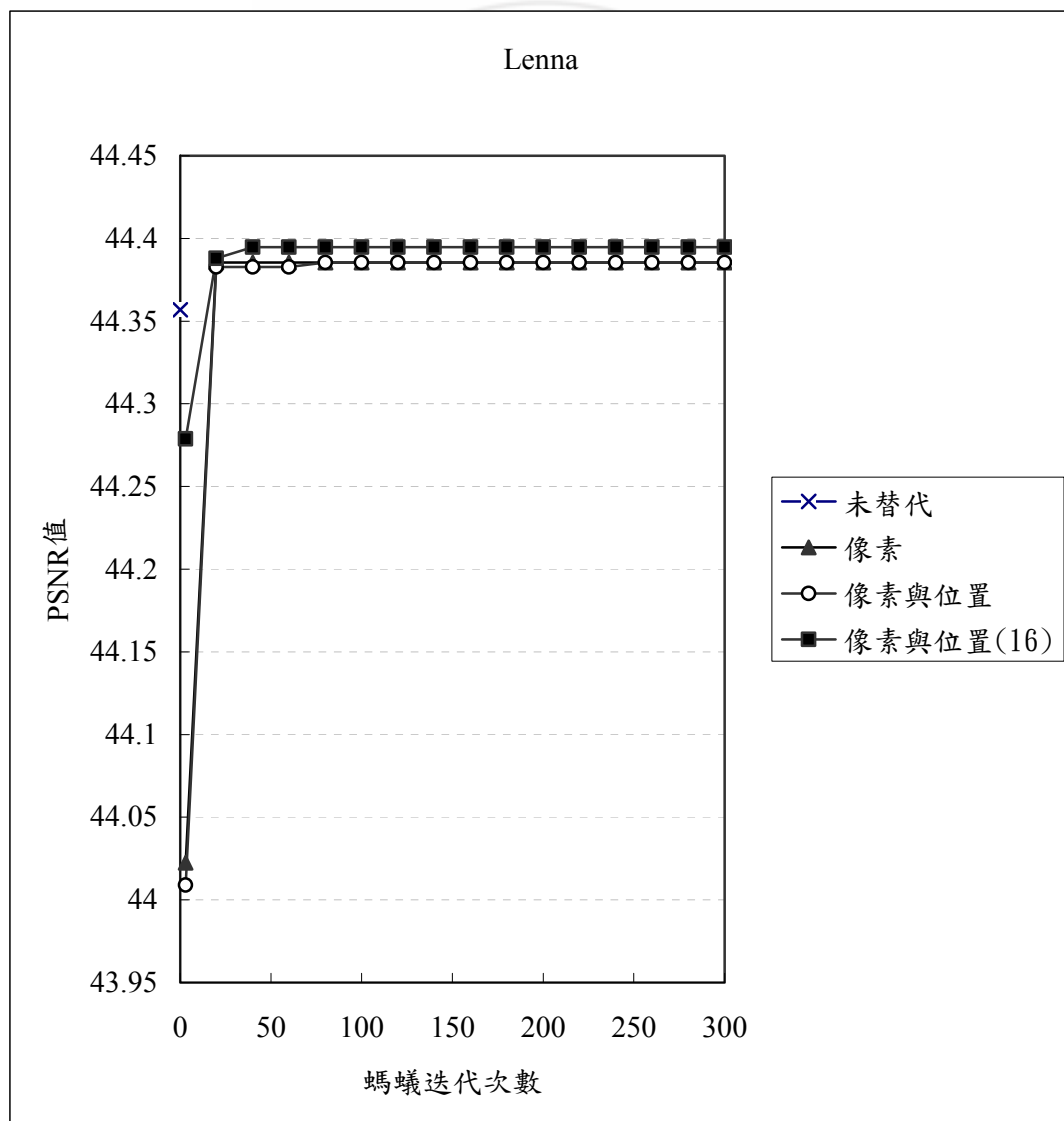


圖 33 Lenna 螞蟻迭代次數與 PSNR 值分佈圖（藏入 2 位元）

10. 掩護影像：Mandarin

表 19 Mandarin 影像介紹（藏 2 位元）

影像類別	掩護影像	秘密訊息
影像名稱	Mandarin	秘密訊息 1
影像大小	512×512	256×256
影像		

表 20 Mandarin 實驗數據（藏 2 位元）

方法	PSNR 值
無替代矩陣	44.351631
4×4 顏色替代矩陣	44.356137
4×4 顏色替代矩陣與 4×4 位置替代矩陣	44.363377
4×4 顏色替代矩陣與 16×16 位置替代矩陣	44.368557

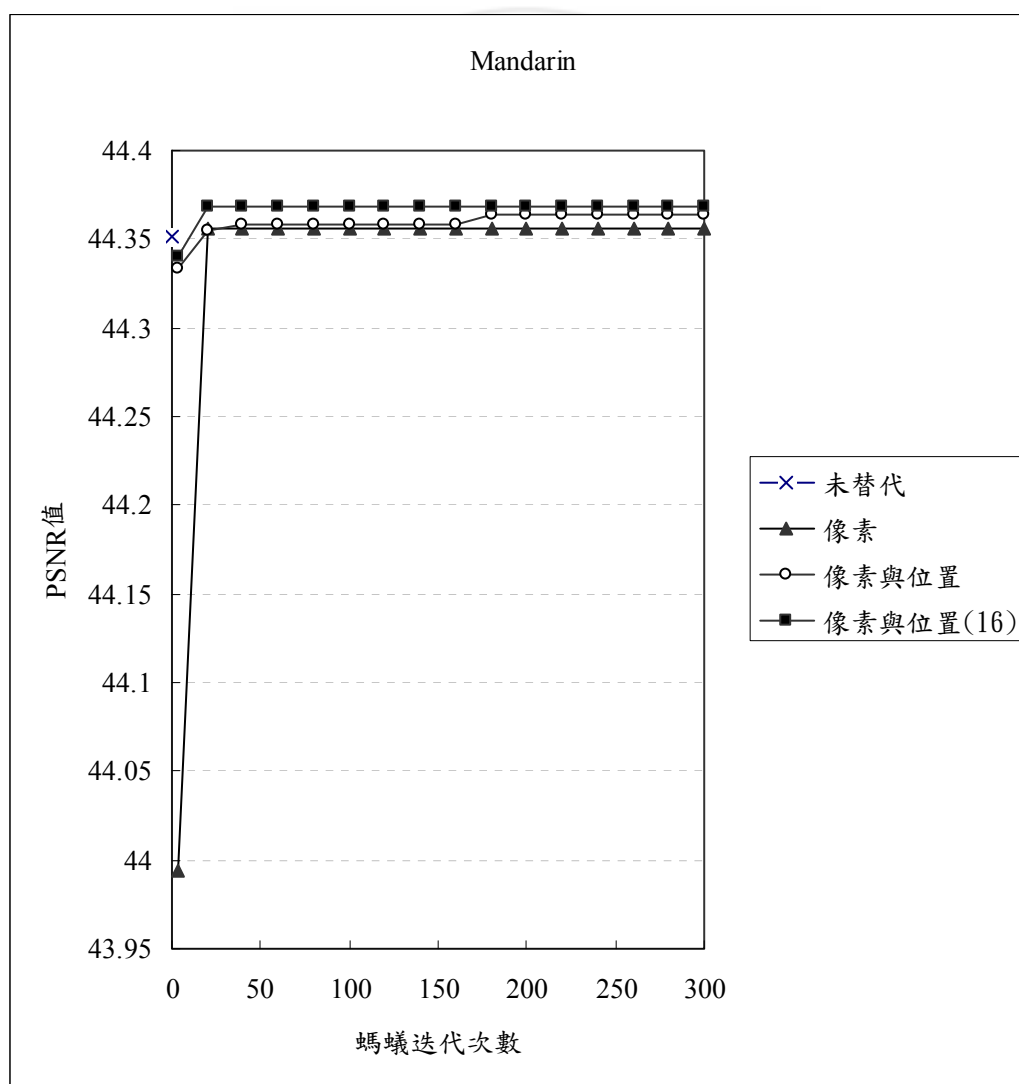


圖 34 Mandarin 螞蟻迭代次數與 PSNR 值分佈圖（藏入 2 位元）

11. 掩護影像：Pepper

表 21 Pepper 影像介紹 (藏 2 位元)

影像類別	掩護影像	秘密訊息
影像名稱	Pepper	秘密訊息 1
影像大小	512×512	256×256
影像		

表 22 Pepper 實驗數據 (藏 2 位元)

方法	PSNR 值
無替代矩陣	44.386414
4×4 顏色替代矩陣	44.387420
4×4 顏色替代矩陣與 4×4 位置替代矩陣	44.398083
4×4 顏色替代矩陣與 16×16 位置替代矩陣	44.403362

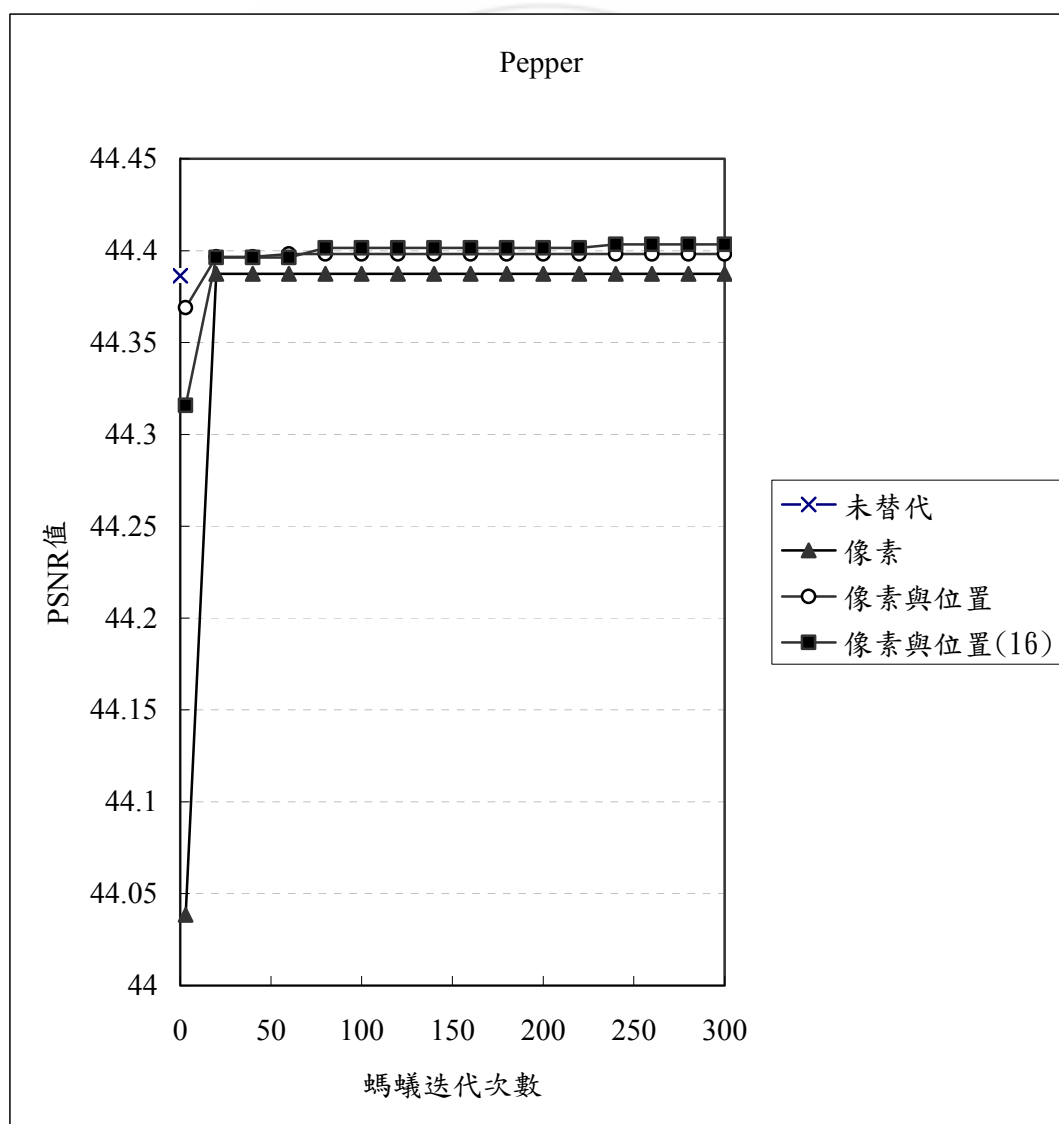


圖 35 Pepper 螞蟻迭代次數與 PSNR 值分佈圖 (藏入 2 位元)

12. 掩護影像：Tiffany

表 23 Tiffany 影像介紹 (藏 2 位元)

影像類別	掩護影像	秘密訊息
影像名稱	Tiffany	秘密訊息 1
影像大小	512×512	256×256
影像		

表 24 Tiffany 實驗數據(藏 2 位元)

方法	PSNR 值
無替代矩陣	44.381081
4×4 顏色替代矩陣	44.382546
4×4 顏色替代矩陣與 4×4 位置替代矩陣	44.406910
4×4 顏色替代矩陣與 16×16 位置替代矩陣	44.419621

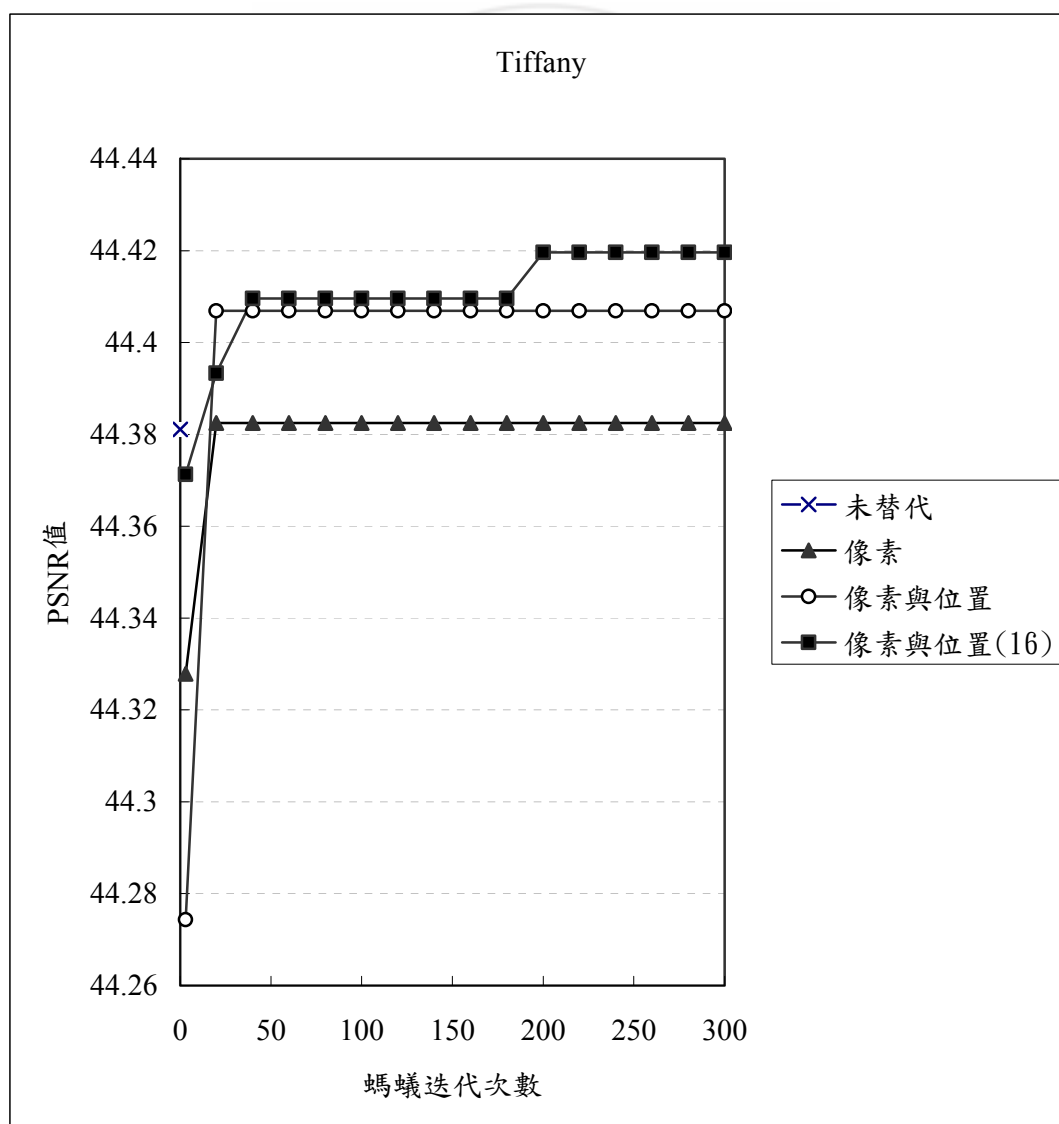


圖 36 Tiffany 螞蟻迭代次數與 PSNR 值分佈圖 (藏入 2 位元)

13. 掩護影像：Tiger

表 25 Tiger 影像介紹（藏 2 位元）

影像類別	掩護影像	秘密訊息
影像名稱	Tiger	秘密訊息 1
影像大小	512×512	256×256
影像		

表 26 Tiger 實驗數據（藏 2 位元）

方法	PSNR 值
無替代矩陣	44.286129
4×4 顏色替代矩陣	44.292160
4×4 顏色替代矩陣與 4×4 位置替代矩陣	44.306835
4×4 顏色替代矩陣與 16×16 位置替代矩陣	44.314091

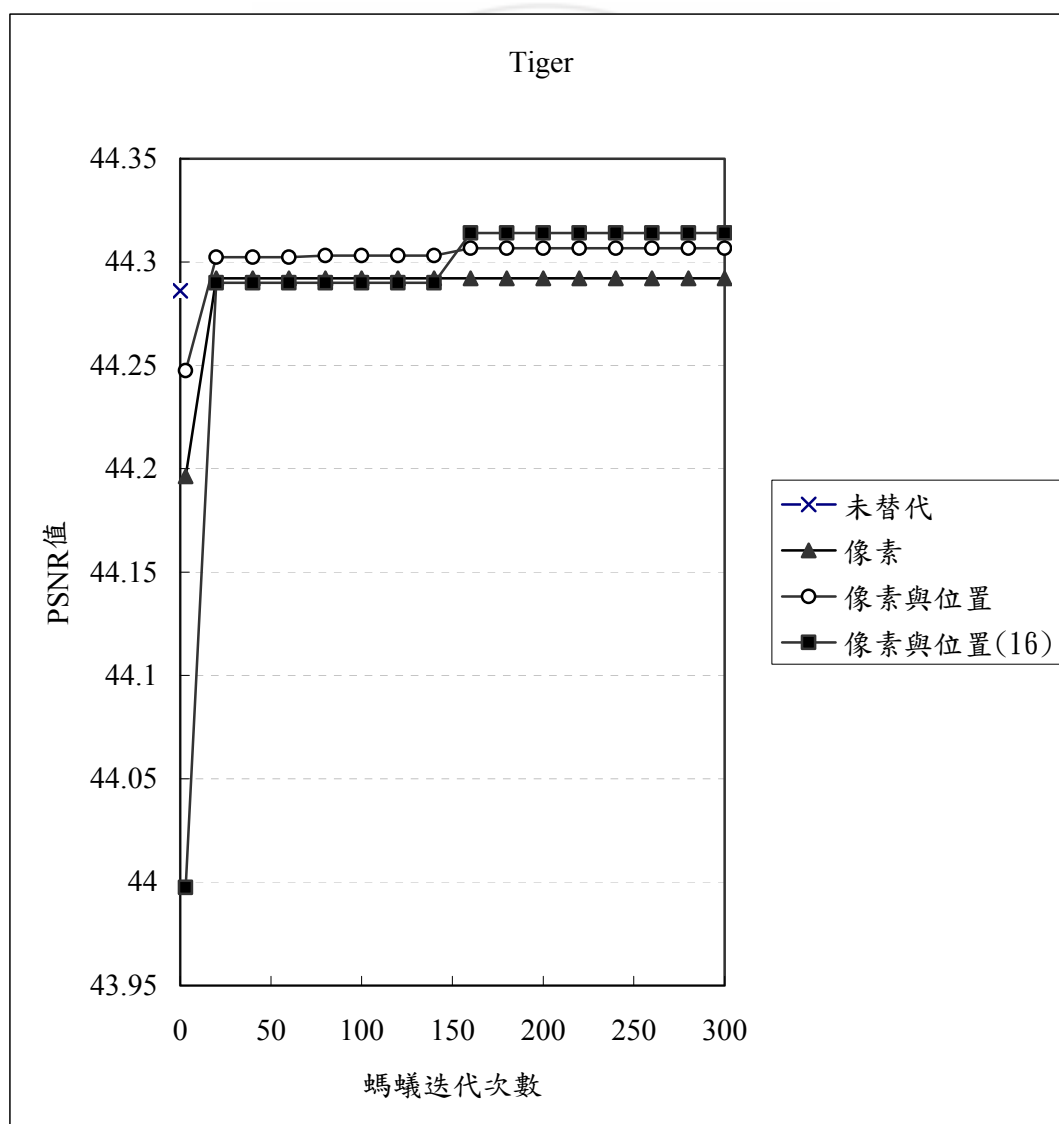


圖 37 Tiger 螞蟻迭代次數與 PSNR 值分佈圖（藏入 2 位元）

14. 掩護影像：Toys

表 27 Toys 影像介紹（藏 2 位元）

影像類別	掩護影像	秘密訊息
影像名稱	Toys	秘密訊息 1
影像大小	512×512	256×256
影像		

表 28 Toys 實驗數據（藏 2 位元）

方法	PSNR 值
無替代矩陣	44.397409
4×4 顏色替代矩陣	44.397409
4×4 顏色替代矩陣與 4×4 位置替代矩陣	44.403923
4×4 顏色替代矩陣與 16×16 位置替代矩陣	44.416878

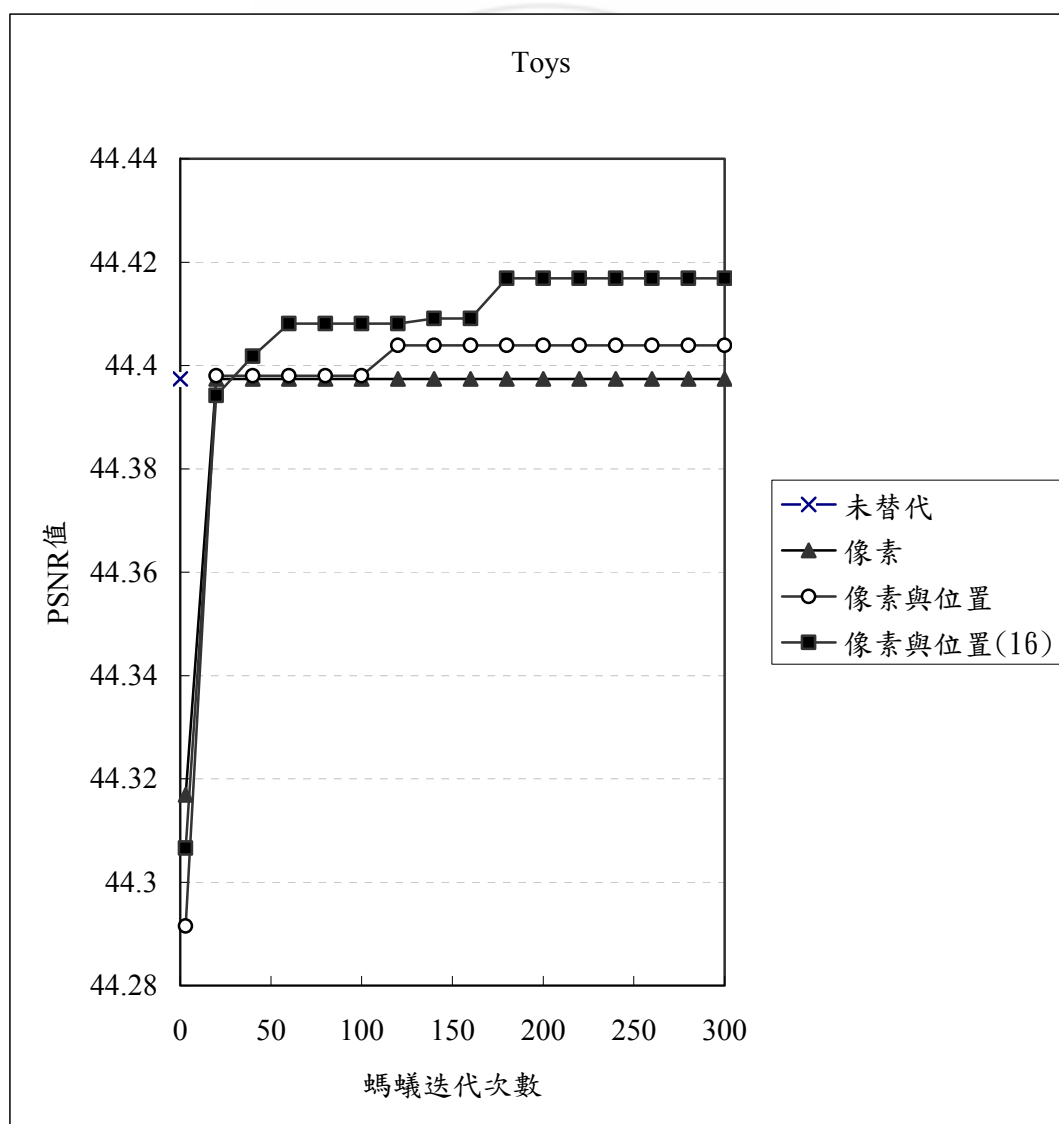


圖 38 Toys 螞蟻迭代次數與 PSNR 值分佈圖（藏入 2 位元）

15. 掩護影像：Zelda

表 29 Zelda 影像介紹（藏 2 位元）

影像類別	掩護影像	秘密訊息
影像名稱	Zelda	秘密訊息 1
影像大小	512×512	256×256
影像		

表 30 Zelda 實驗數據（藏 2 位元）

方法	PSNR 值
無替代矩陣	44.384617
4×4 顏色替代矩陣	44.391193
4×4 顏色替代矩陣與 4×4 位置替代矩陣	44.402397
4×4 顏色替代矩陣與 16×16 位置替代矩陣	44.407764

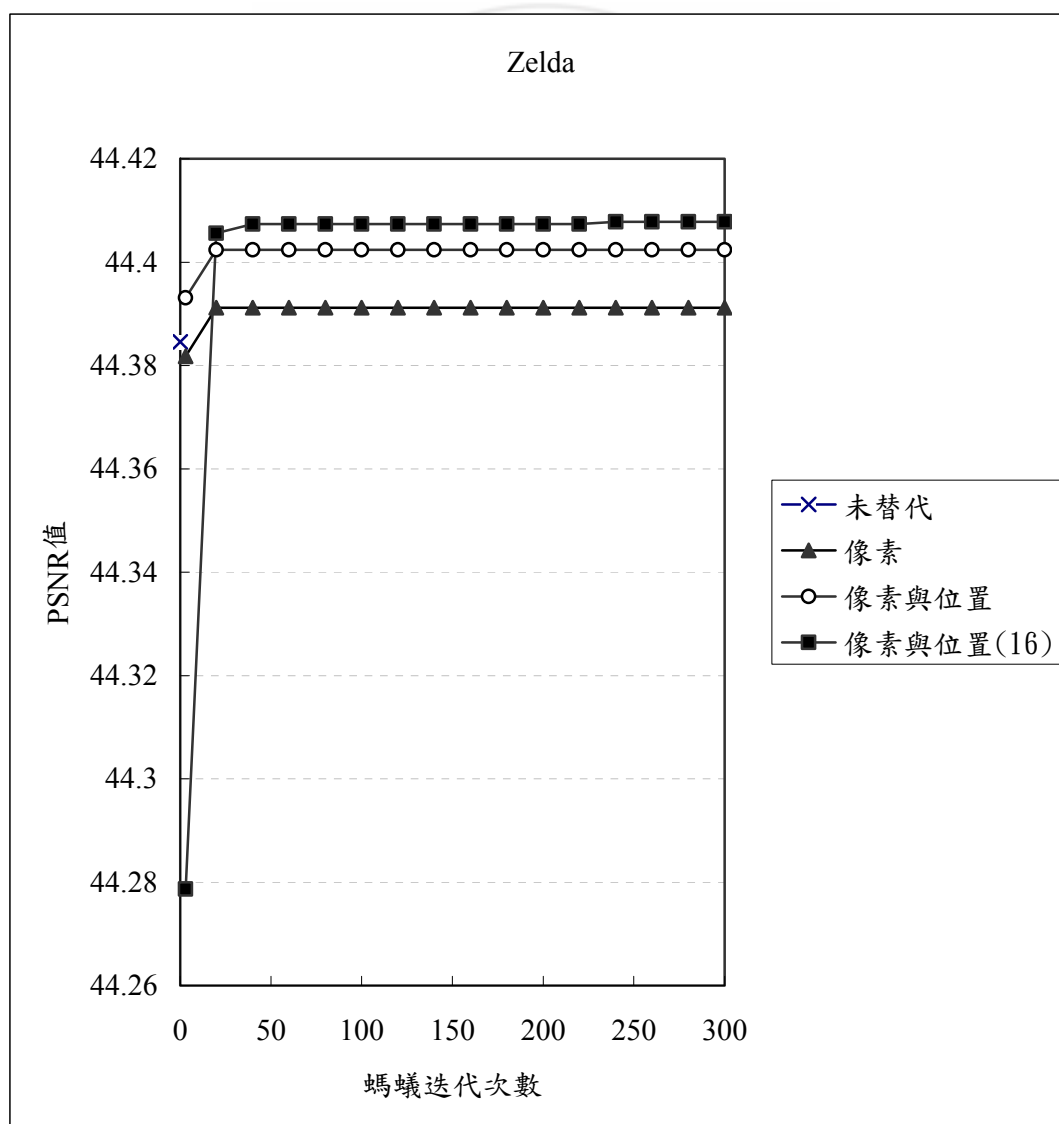


圖 39 Zelda 螞蟻迭代次數與 PSNR 值分佈圖（藏入 2 位元）

16. 掩護影像：銘傳大學

表 31 銘傳大學影像介紹（藏 2 位元）

影像類別	掩護影像	秘密訊息
影像名稱	銘傳大學	秘密訊息 1
影像大小	512×512	256×256
影像		

表 32 銘傳大學實驗數據（藏 2 位元）

方法	PSNR 值
無替代矩陣	44.803741
4×4 顏色替代矩陣	44.975260
4×4 顏色替代矩陣與 4×4 位置替代矩陣	44.985329
4×4 顏色替代矩陣與 16×16 位置替代矩陣	44.993496

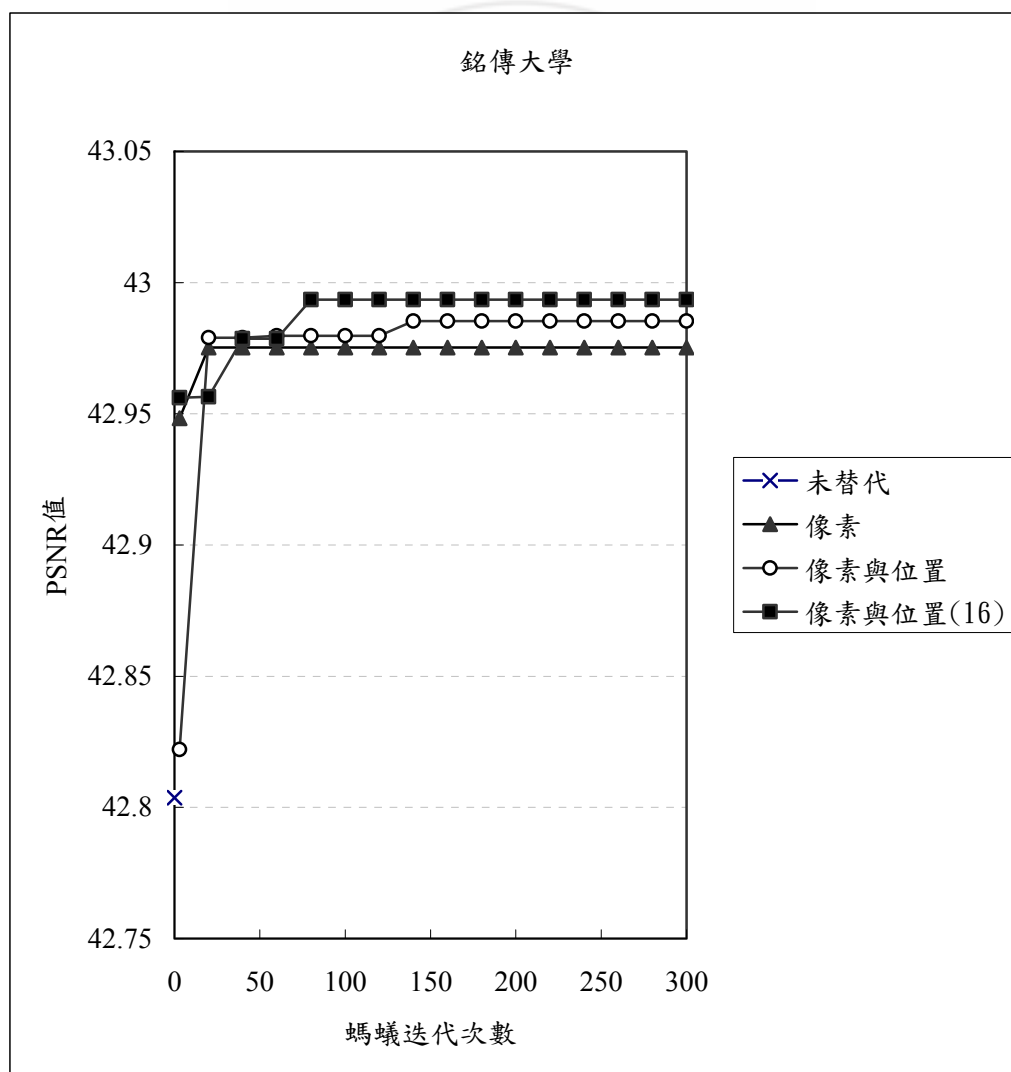


圖 40 銘傳大學螞蟻迭代次數與 PSNR 值分佈圖（藏入 2 位元）

17. 掩護影像：銘傳資管

表 33 銘傳資管影像介紹（藏 2 位元）



影像類別	掩護影像	秘密訊息
影像名稱	銘傳資管	秘密訊息 1
影像大小	512×512	256×256
影像		

表 34 銘傳資管實驗數據（藏 2 位元）

方法	PSNR 值
無替代矩陣	44.661274
4×4 顏色替代矩陣	44.661274
4×4 顏色替代矩陣與 4×4 位置替代矩陣	44.661274
4×4 顏色替代矩陣與 16×16 位置替代矩陣	44.678616

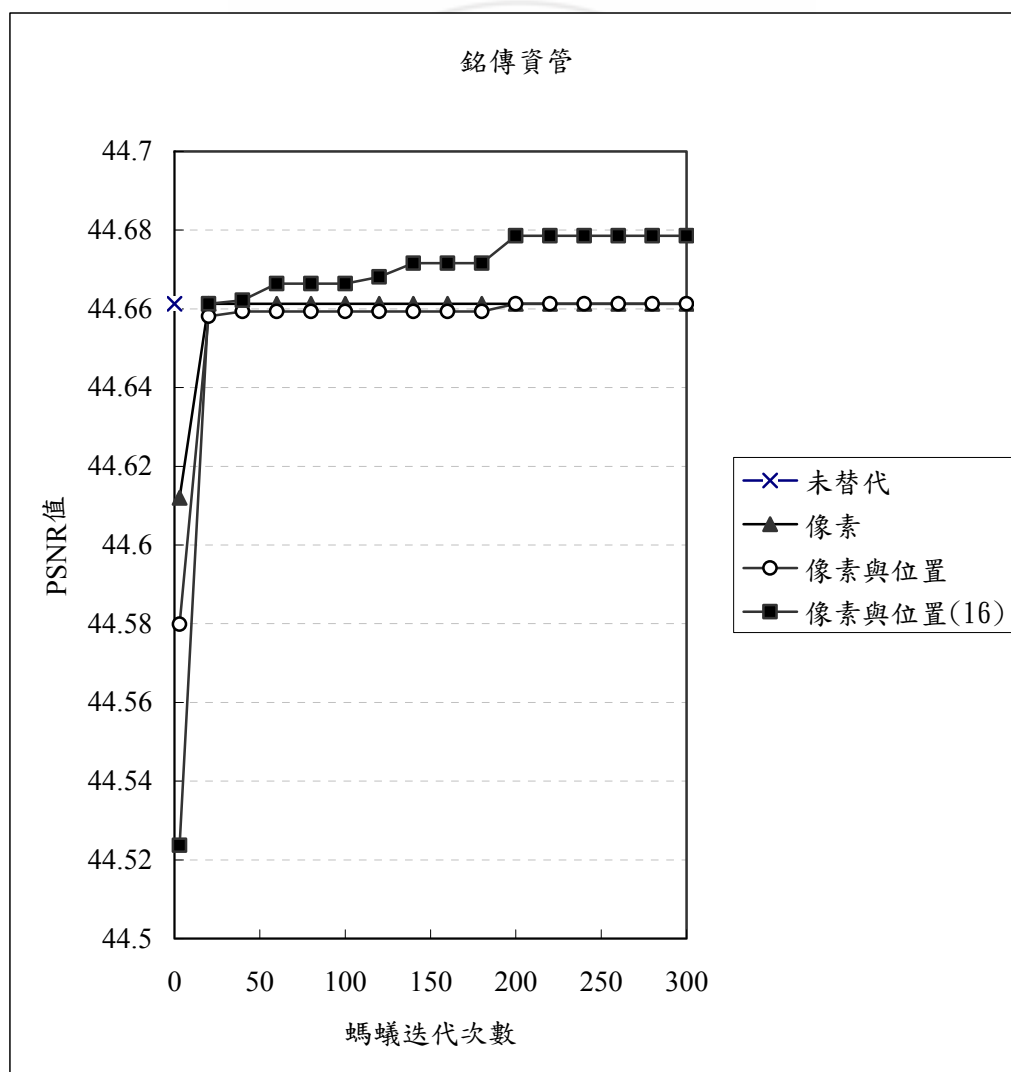


圖 41 銘傳資管螞蟻迭代次數與 PSNR 值分佈圖（藏入 2 位元）

4.3 掩護影像的每個像素藏入 4 位元秘密訊息的實驗結果

在本節中我們將呈現每張掩護影像的每個像素皆藏入 4 位元的秘密訊息的實驗結果。我們總共分成三種不同的實驗，第一種實驗是未使用任何的替代矩陣來藏入秘密訊息；第二種實驗是只使用 16×16 的顏色替代矩陣來轉變秘密訊息；第三種實驗是使用具有相互關係的 16×16 的顏色替代矩陣與 16×16 的位置替代矩陣來轉變秘密訊息。

在 4.3 的實驗中，我們分別使用 18 張不同的掩護影像，大小為 512×512 來藏入 2 種不同的秘密訊息，大小為 256×512 ，實驗結果如下列各表所示：



1. 掩護影像：Airplane

表 35 Airplane 影像介紹 (藏 4 位元)

影像類別	掩護影像	秘密訊息
影像名稱	Airplane	秘密訊息 1
影像大小	512×512	256×512
影像		

表 36 Airplane 實驗數據 (藏 4 位元)

方法	PSNR 值
無替代矩陣	32.340786
16×16 顏色替代矩陣	32.516765
16×16 顏色替代矩陣與 16×16 位置替代矩陣	32.527241

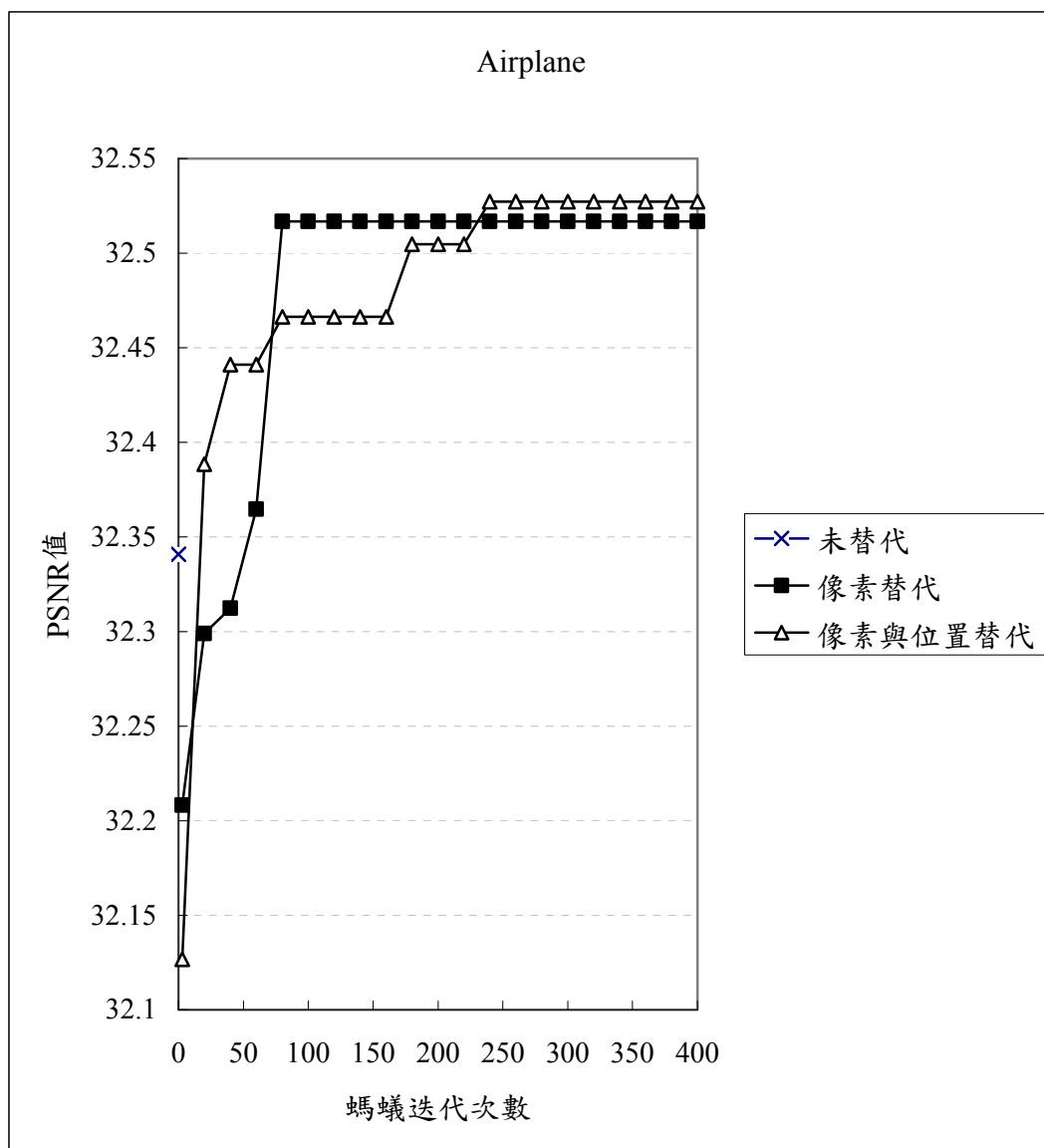


圖 42 Airplane 螞蟻迭代次數與 PSNR 值分佈圖 (藏入 4 位元)

2. 掩護影像: Baboon

表 37 Baboon 影像介紹 (藏 4 位元)



影像類別	掩護影像	秘密訊息
影像名稱	Baboon	秘密訊息 1
影像大小	512×512	256×512
影像		

表 38 Baboon 實驗數據 (藏 4 位元)

方法	PSNR 值
無替代矩陣	32.406961
16×16 顏色替代矩陣	32.520500
16×16 顏色替代矩陣與 16×16 位置替代矩陣	32.546455

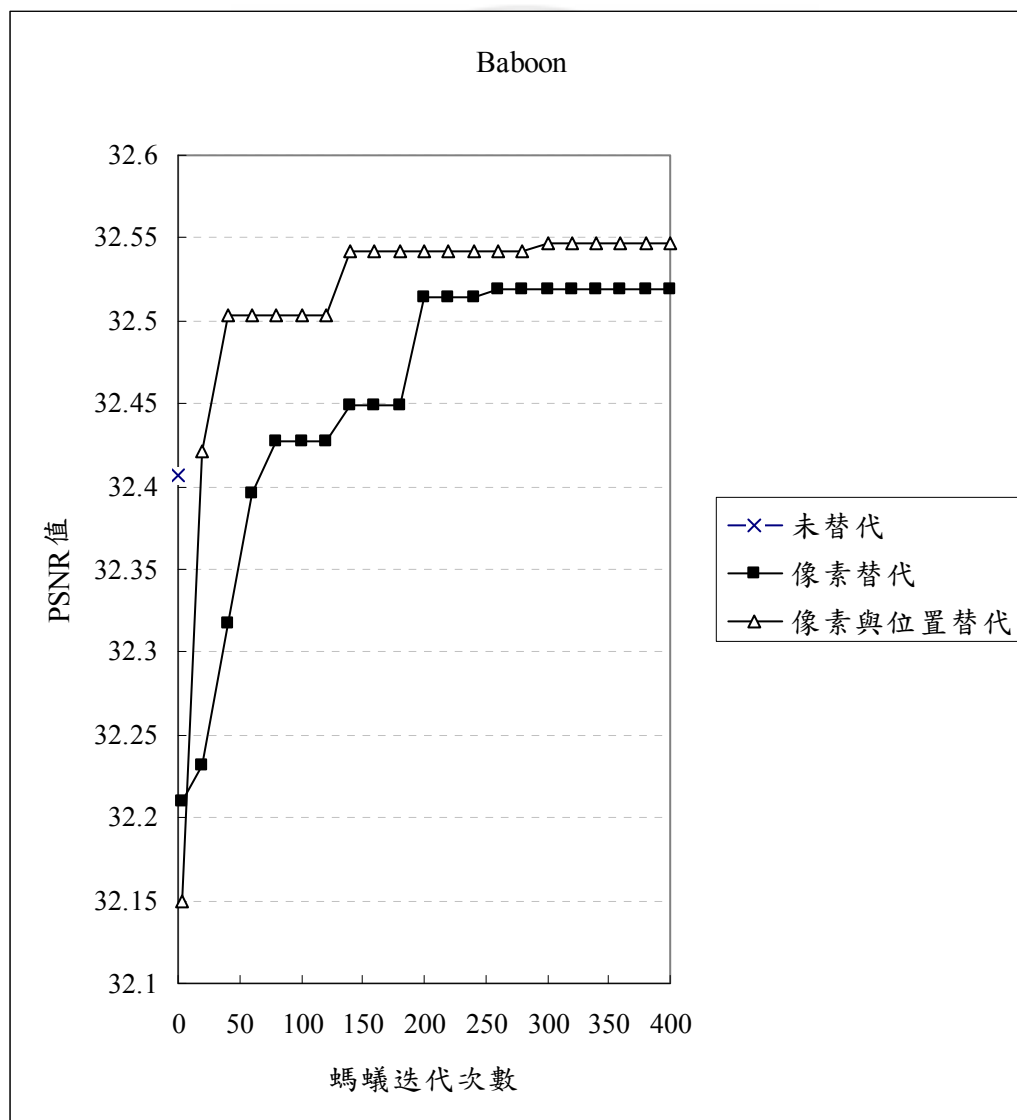


圖 43 Baboon 螞蟻迭代次數與 PSNR 值分佈圖 (藏入 4 位元)

3. 掩護影像: Bird

表 39 Bird 影像介紹 (藏 4 位元)

影像類別	掩護影像	秘密訊息
影像名稱	Bird	秘密訊息 1
影像大小	512×512	256×512
影像		

表 40 Bird 實驗數據 (藏 4 位元)

方法	PSNR 值
無替代矩陣	32.367825
16×16 顏色替代矩陣	32.464263
16×16 顏色替代矩陣與 16×16 位置替代矩陣	32.504772

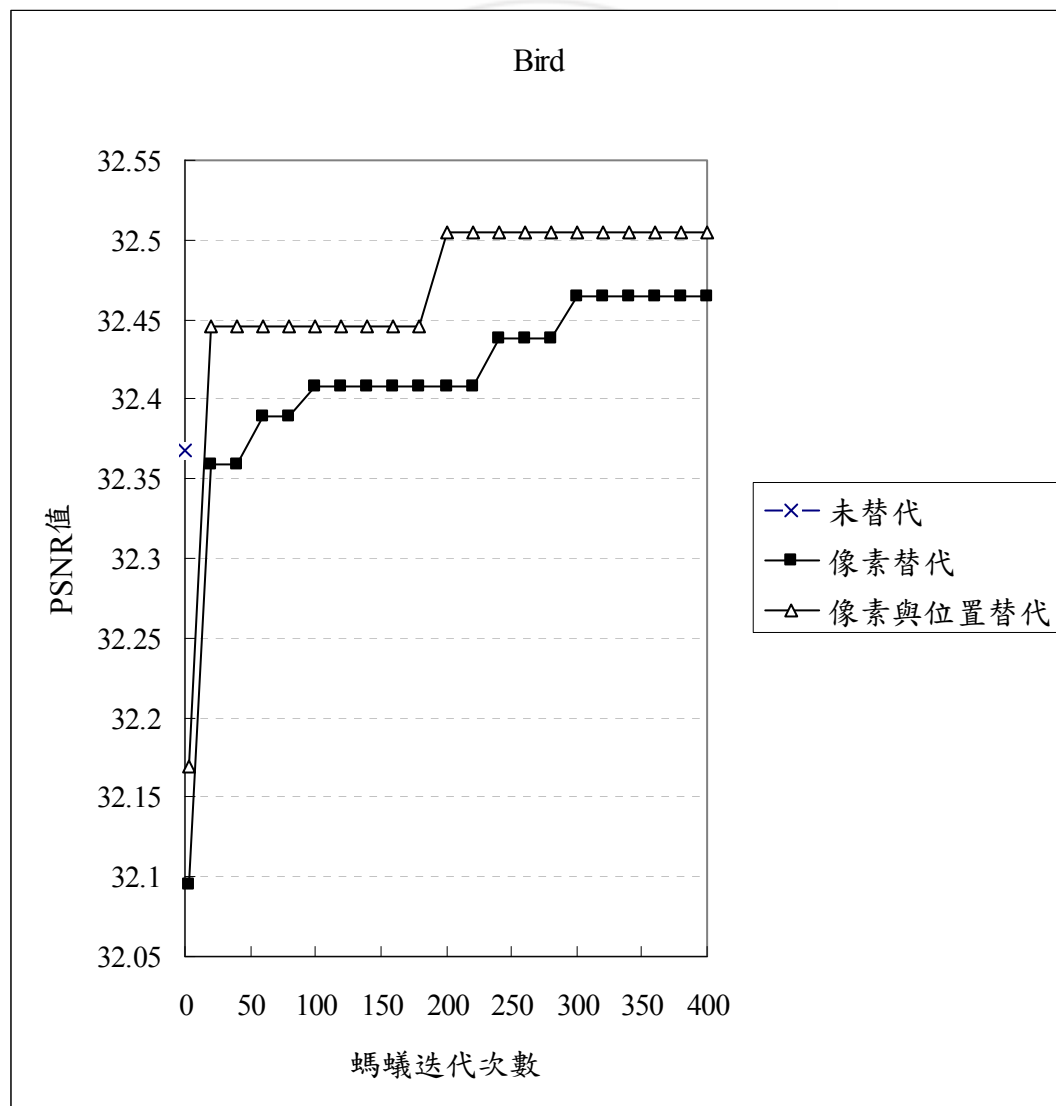


圖 44 Bird 螞蟻迭代次數與 PSNR 值分佈圖 (藏入 4 位元)

4. 掩護影像: Bird1

表 41 Bird1 影像介紹 (藏 4 位元)

影像類別	掩護影像	秘密訊息
影像名稱	Bird1	秘密訊息 1
影像大小	512×512	256×512
影像		

表 42 Bird1 實驗數據 (藏 4 位元)

方法	PSNR 值
無替代矩陣	32.795255
16×16 顏色替代矩陣	32.811638
16×16 顏色替代矩陣與 16×16 位置替代矩陣	32.861458

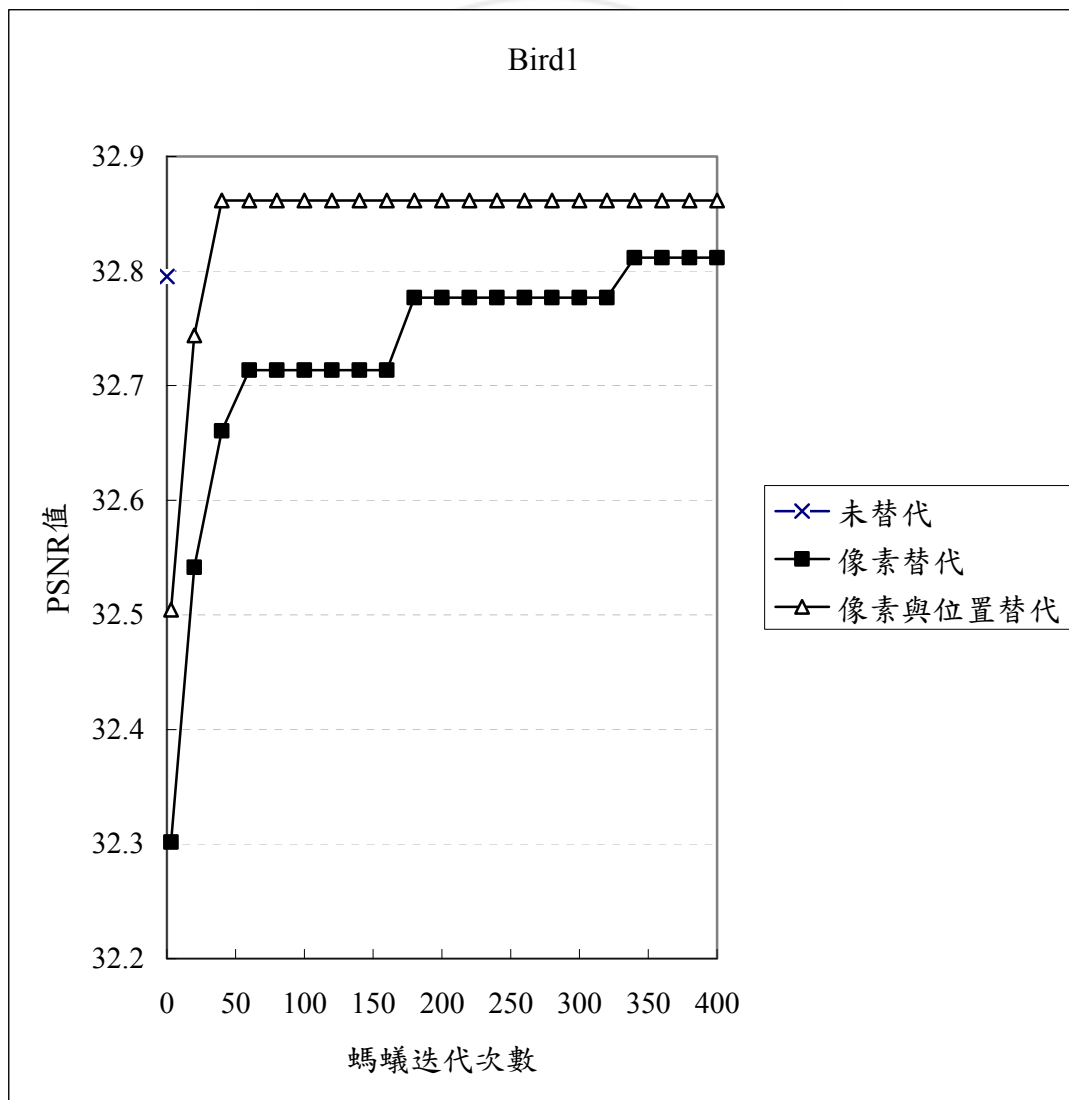


圖 45 Bird1 螞蟻迭代次數與 PSNR 值分佈圖 (藏入 4 位元)

5. 掩護影像: Boat

表 43 Boat 影像介紹 (藏 4 位元)

影像類別	掩護影像	秘密訊息
影像名稱	Boat	秘密訊息 1
影像大小	512×512	256×512
影像		

表 44 Boat 實驗數據 (藏 4 位元)

方法	PSNR 值
無替代矩陣	32.310890
16×16 顏色替代矩陣	32.421171
16×16 顏色替代矩陣與 16×16 位置替代矩陣	32.451656

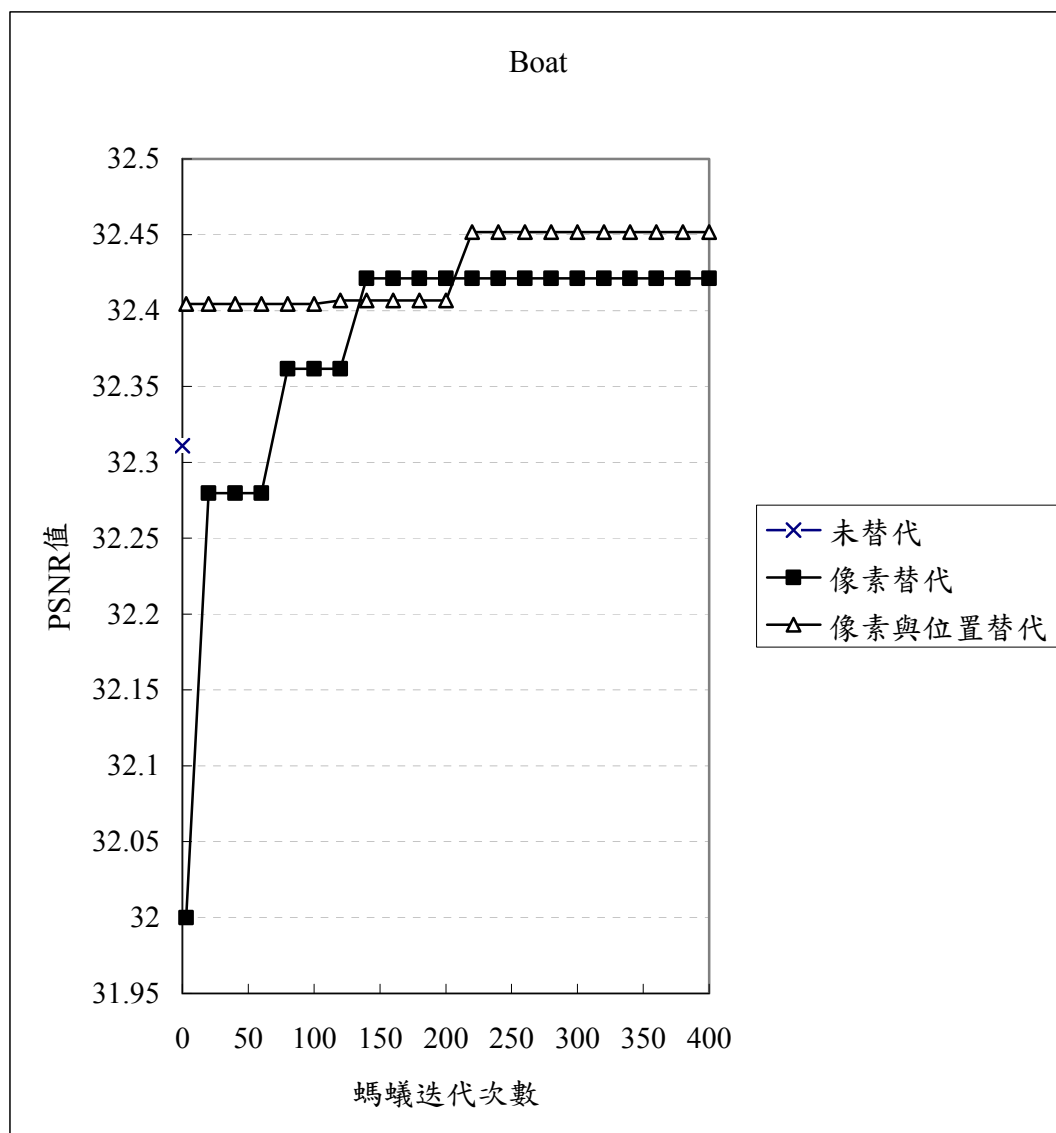


圖 46 Boat 螞蟻迭代次數與 PSNR 值分佈圖 (藏入 4 位元)

6. 掩護影像:Cat

表 45 Cat 影像介紹 (藏 4 位元)

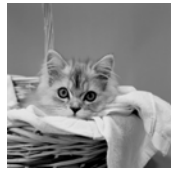

影像類別	掩護影像	秘密訊息
影像名稱	Cat	秘密訊息 1
影像大小	512×512	256×512
影像		

表 46 Cat 實驗數據 (藏 4 位元)

方法	PSNR 值
無替代矩陣	32.448398
16×16 顏色替代矩陣	32.540310
16×16 顏色替代矩陣與 16×16 位置替代矩陣	32.593388

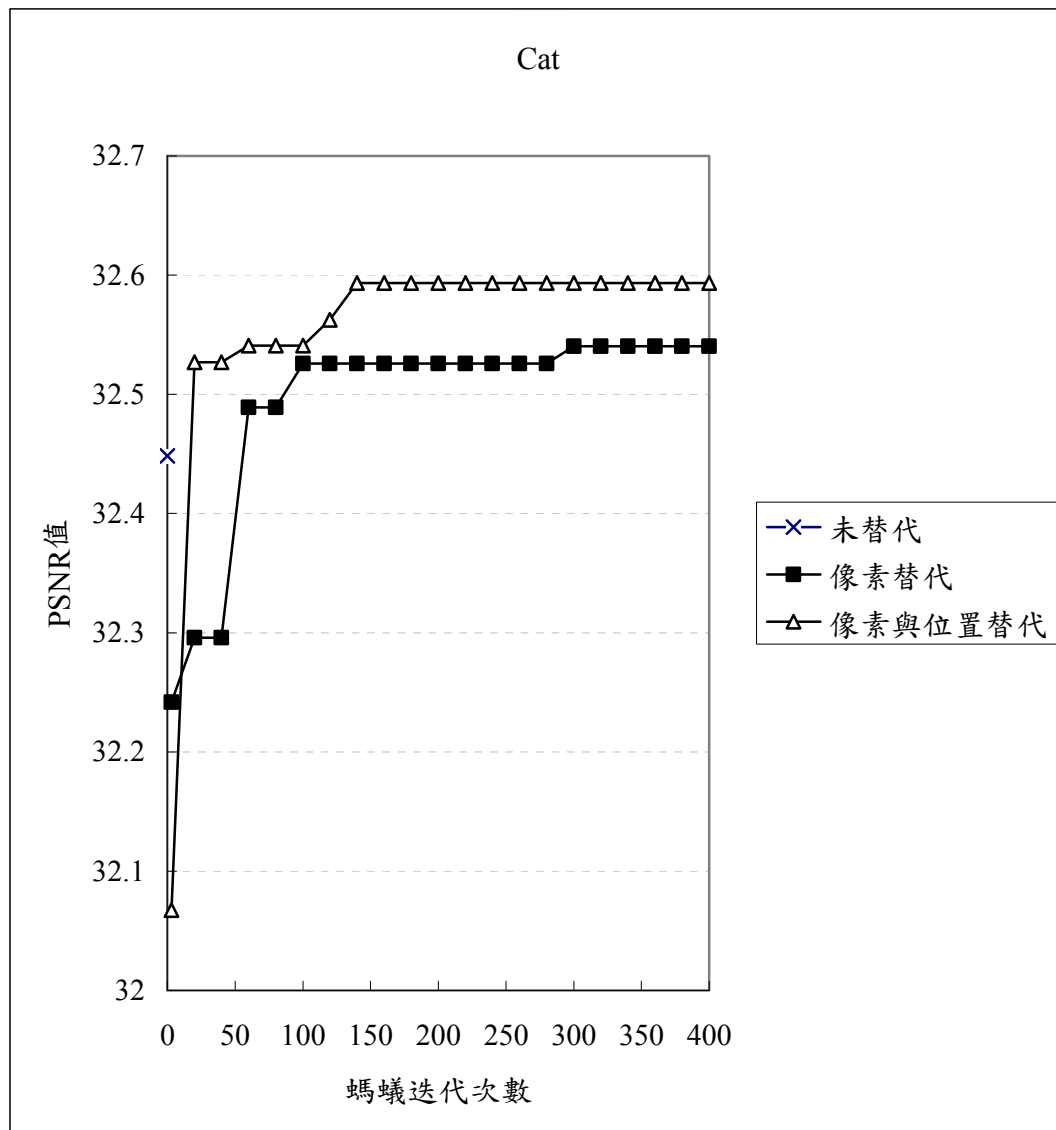


圖 47 Cat 螞蟻迭代次數與 PSNR 值分佈圖 (藏入 4 位元)

7. 掩護影像:Girl

表 47 Girl 影像介紹 (藏 4 位元)



影像類別	掩護影像	秘密訊息
影像名稱	Girl	秘密訊息 1
影像大小	512×512	256×512
影像		

表 48 Girl 實驗數據 (藏 4 位元)

方法	PSNR 值
無替代矩陣	32.288845
16×16 顏色替代矩陣	32.406542
16×16 顏色替代矩陣與 16×16 位置替代矩陣	32.417171

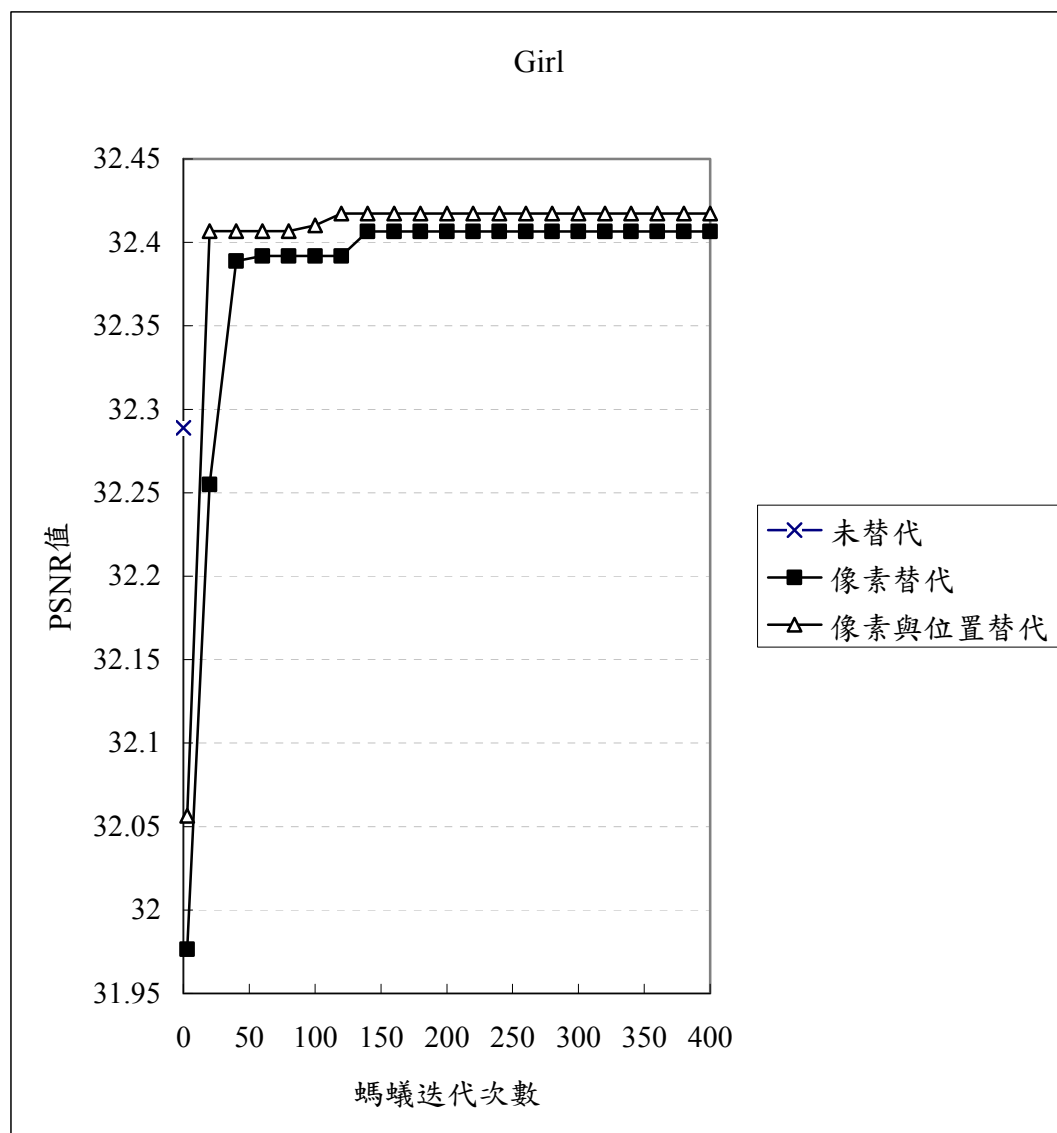


圖 48 Girl 螞蟻迭代次數與 PSNR 值分佈圖 (藏入 4 位元)

8. 掩護影像:Gold

表 49 Gold 影像介紹 (藏 4 位元)



影像類別	掩護影像	秘密訊息
影像名稱	Gold	秘密訊息 1
影像大小	512×512	256×512
影像		

表 50 Gold 實驗數據 (藏 4 位元)

方法	PSNR 值
無替代矩陣	32.420765
16×16 顏色替代矩陣	32.464535
16×16 顏色替代矩陣與 16×16 位置替代矩陣	32.546227

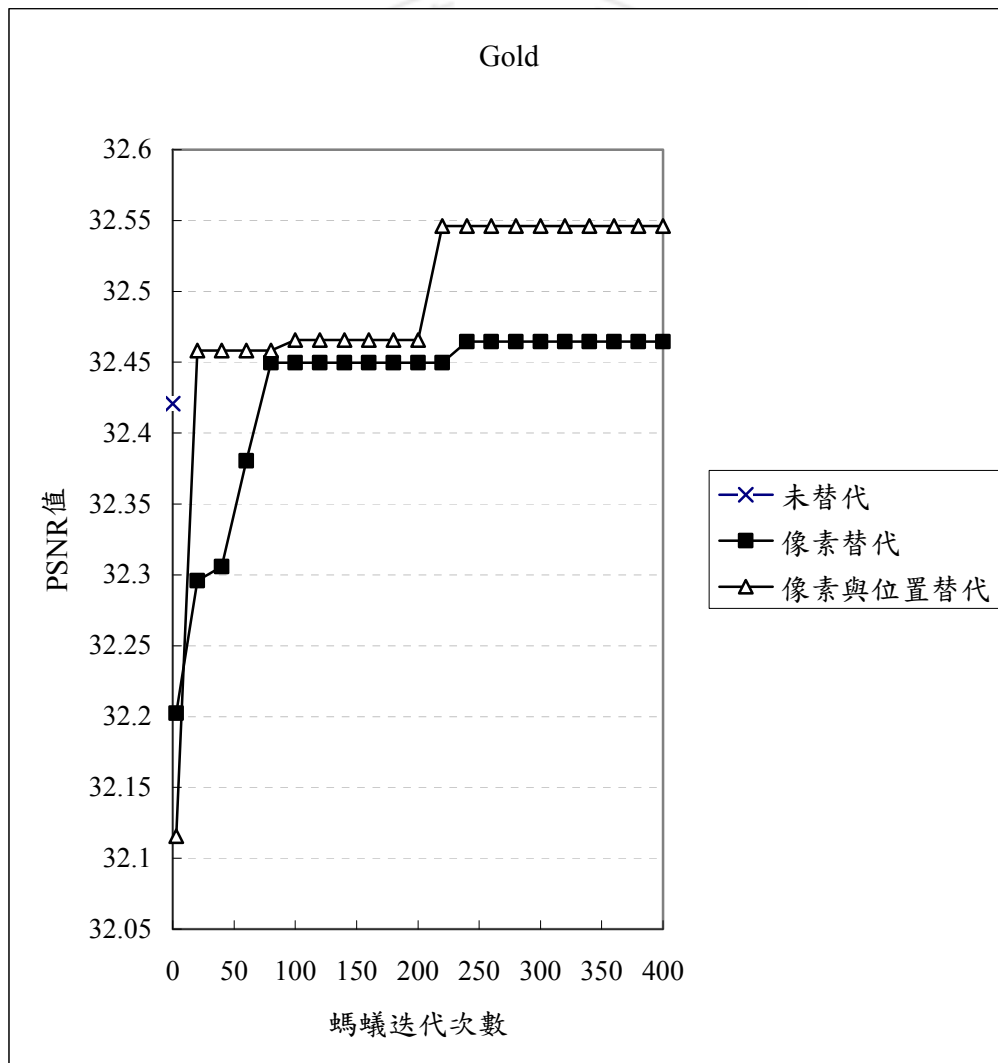


圖 49 Gold 螞蟻迭代次數與 PSNR 值分佈圖 (藏入 4 位元)

9. 掩護影像：Lena

表 51 Lena 影像介紹（藏 4 位元）



影像類別	掩護影像	秘密訊息
影像名稱	Lena	秘密訊息 1
影像大小	512×512	256×512
影像		

表 52 Lena 實驗數據（藏 4 位元）

方法	PSNR 值
無替代矩陣	32.381073
16×16 顏色替代矩陣	32.490992
16×16 顏色替代矩陣與 16×16 位置替代矩陣	32.503159

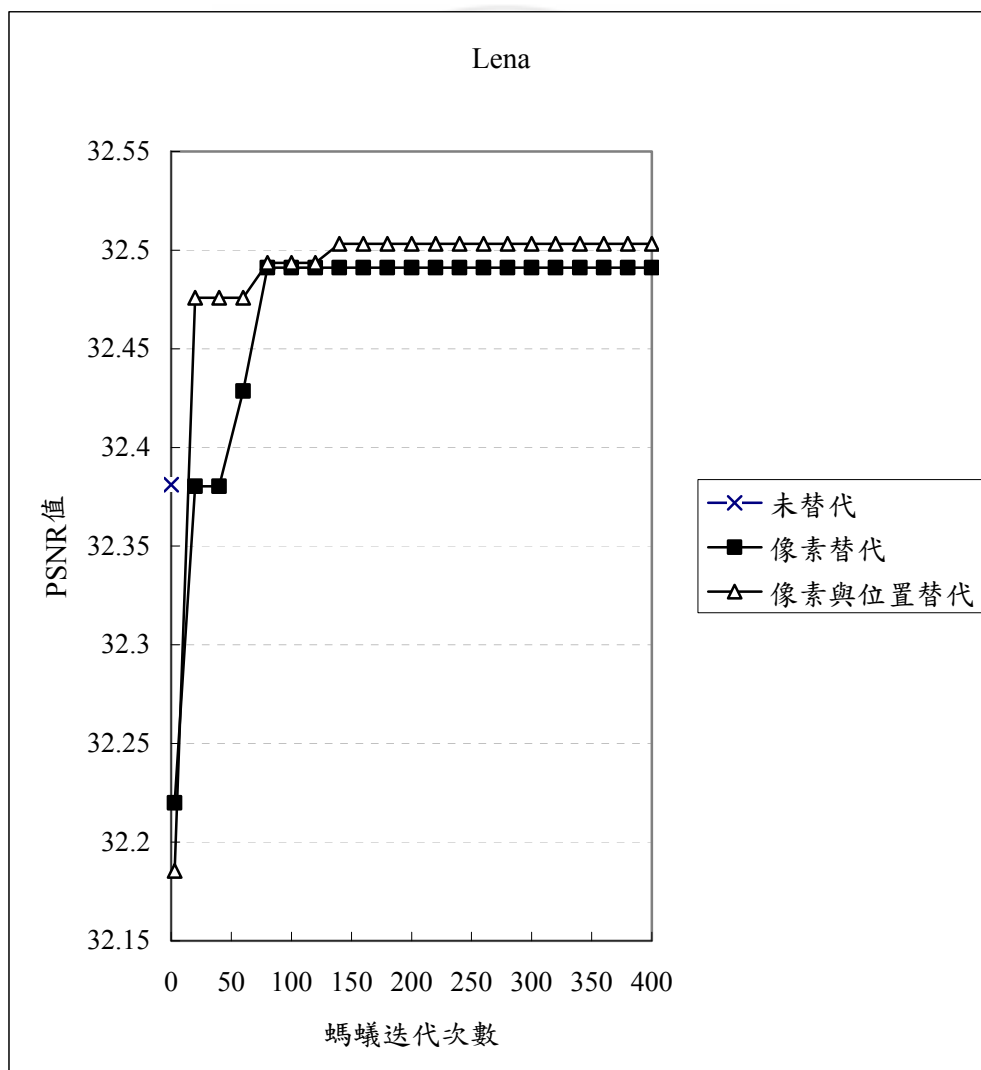


圖 50 Lena 螞蟻迭代次數與 PSNR 值分佈圖（藏入 4 位元）

10. 影像:Lenna

表 53 Lenna 影像介紹 (藏 4 位元)



影像類別	掩護影像	秘密訊息
影像名稱	Lenna	秘密訊息 1
影像大小	512×512	256×512
影像		

表 54 Lenna 實驗數據 (藏 4 位元)

方法	PSNR 值
無替代矩陣	32.381439
16×16 顏色替代矩陣	32.483063
16×16 顏色替代矩陣與 16×16 位置替代矩陣	32.505615

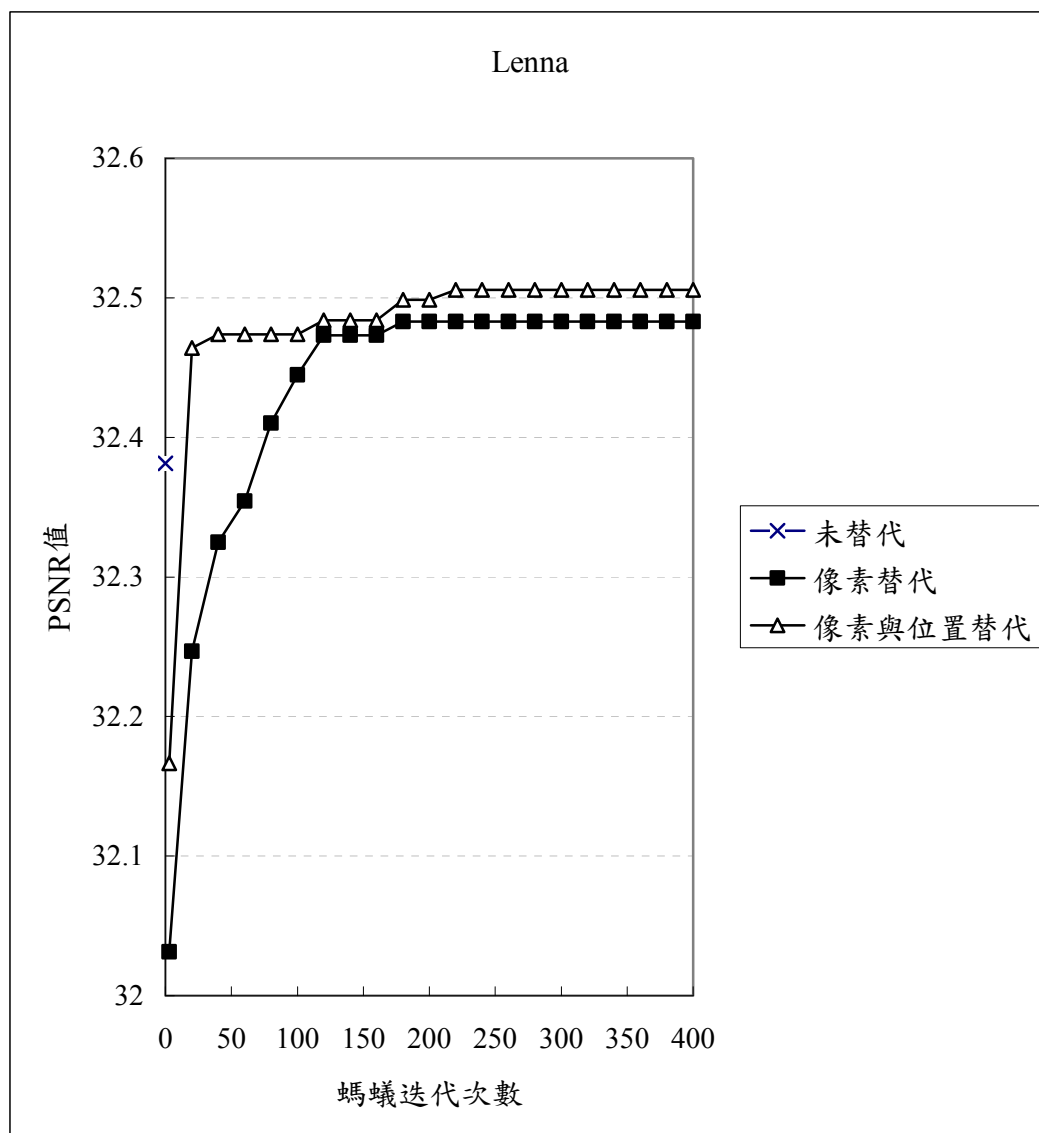


圖 51 Lenna 螞蟻迭代次數與 PSNR 值分佈圖 (藏入 4 位元)

11. 掩護影像:Monalisa

表 55 Monalisa 影像介紹 (藏 4 位元)

影像類別	掩護影像	秘密訊息
影像名稱	Monalisa	秘密訊息 1
影像大小	512×512	256×512
影像		

表 56 Monalisa 實驗數據(藏 4 位元)

方法	PSNR 值
無替代矩陣	32.556849
16×16 顏色替代矩陣	32.648446
16×16 顏色替代矩陣與 16×16 位置替代矩陣	32.674183

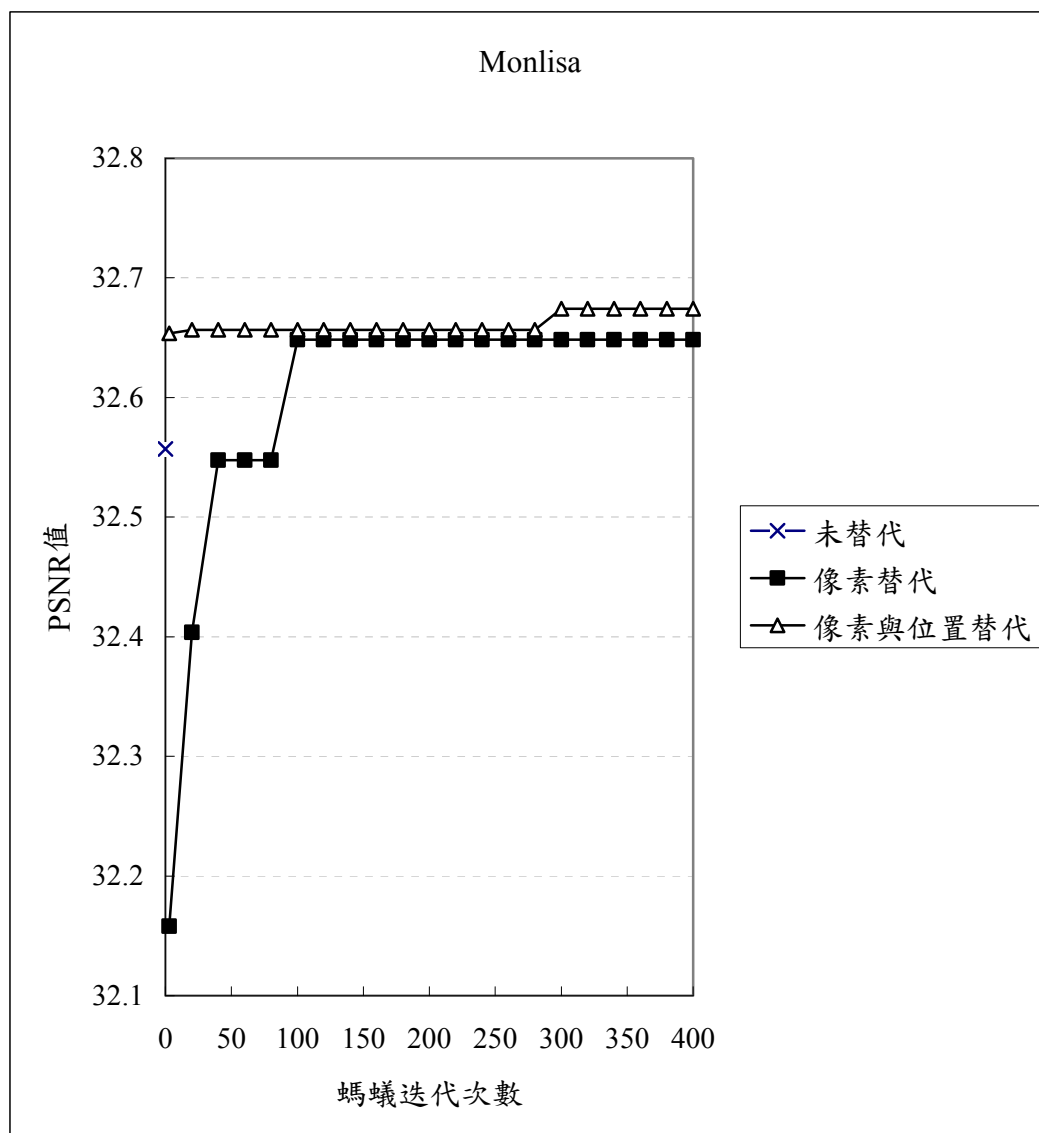


圖 52 Monalisa 螞蟻迭代次數與 PSNR 值分佈圖 (藏入 4 位元)

12. 掩護影像:Sailboat

表 57 Sailboat 影像介紹 (藏 4 位元)



影像類別	掩護影像	秘密訊息
影像名稱	Sailboat	秘密訊息 1
影像大小	512×512	256×512
影像		

表 58 Sailboat 實驗數據 (藏 4 位元)

方法	PSNR 值
無替代矩陣	32.283610
16×16 顏色替代矩陣	32.341539
16×16 顏色替代矩陣與 16×16 位置替代矩陣	32.378742

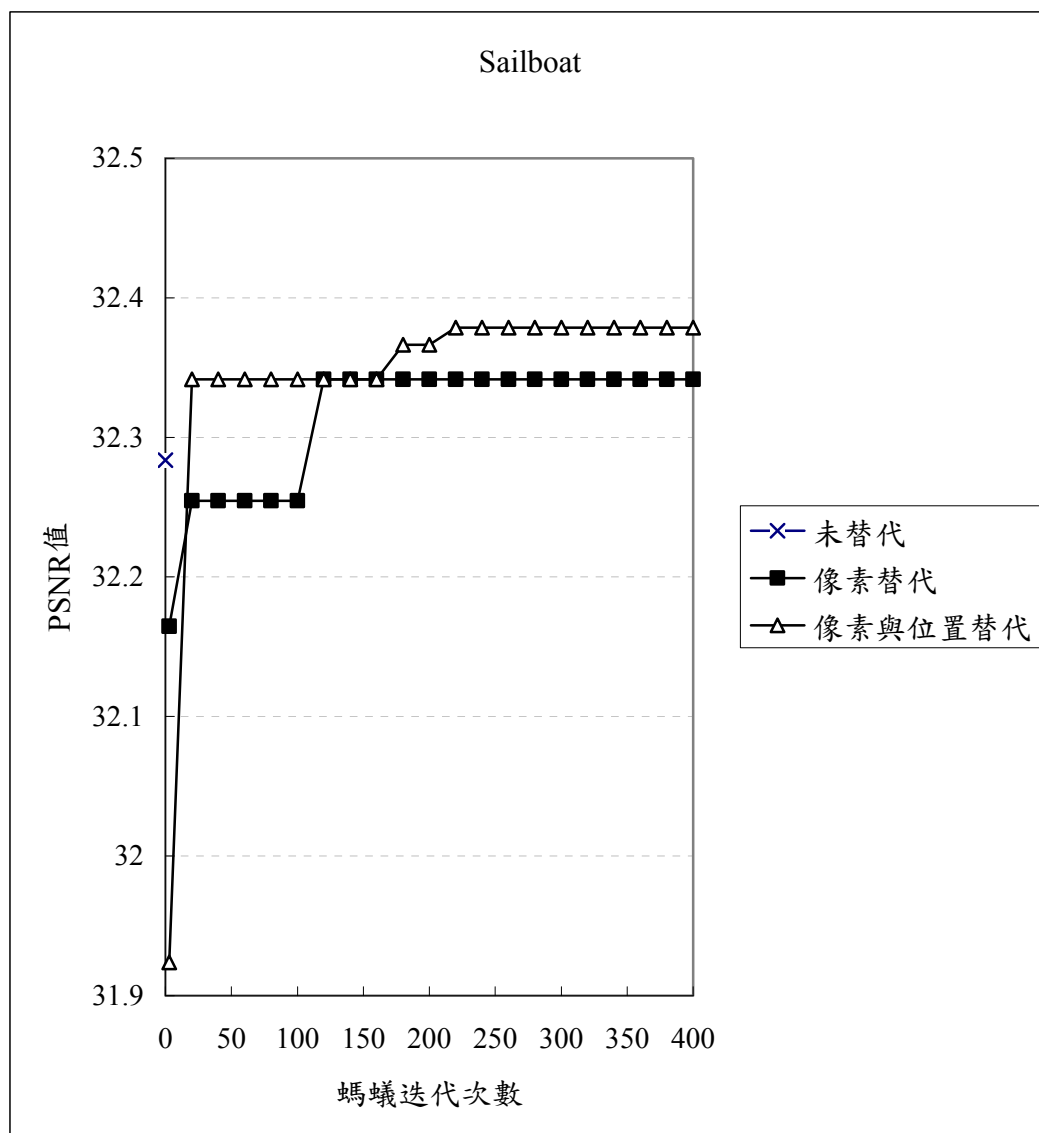


圖 53 Sailboat 螞蟻迭代次數與 PSNR 值分佈圖 (藏入 4 位元)

13. 掩護影像:Tiffany

表 59 Tiffany 影像介紹 (藏 4 位元)

影像類別	掩護影像	秘密訊息
影像名稱	Tiffany	秘密訊息 1
影像大小	512×512	256×512
影像		

表 60 Tiffany 實驗數據 (藏 4 位元)

方法	PSNR 值
無替代矩陣	32.493758
16×16 顏色替代矩陣	32.569562
16×16 顏色替代矩陣與 16×16 位置替代矩陣	32.615524

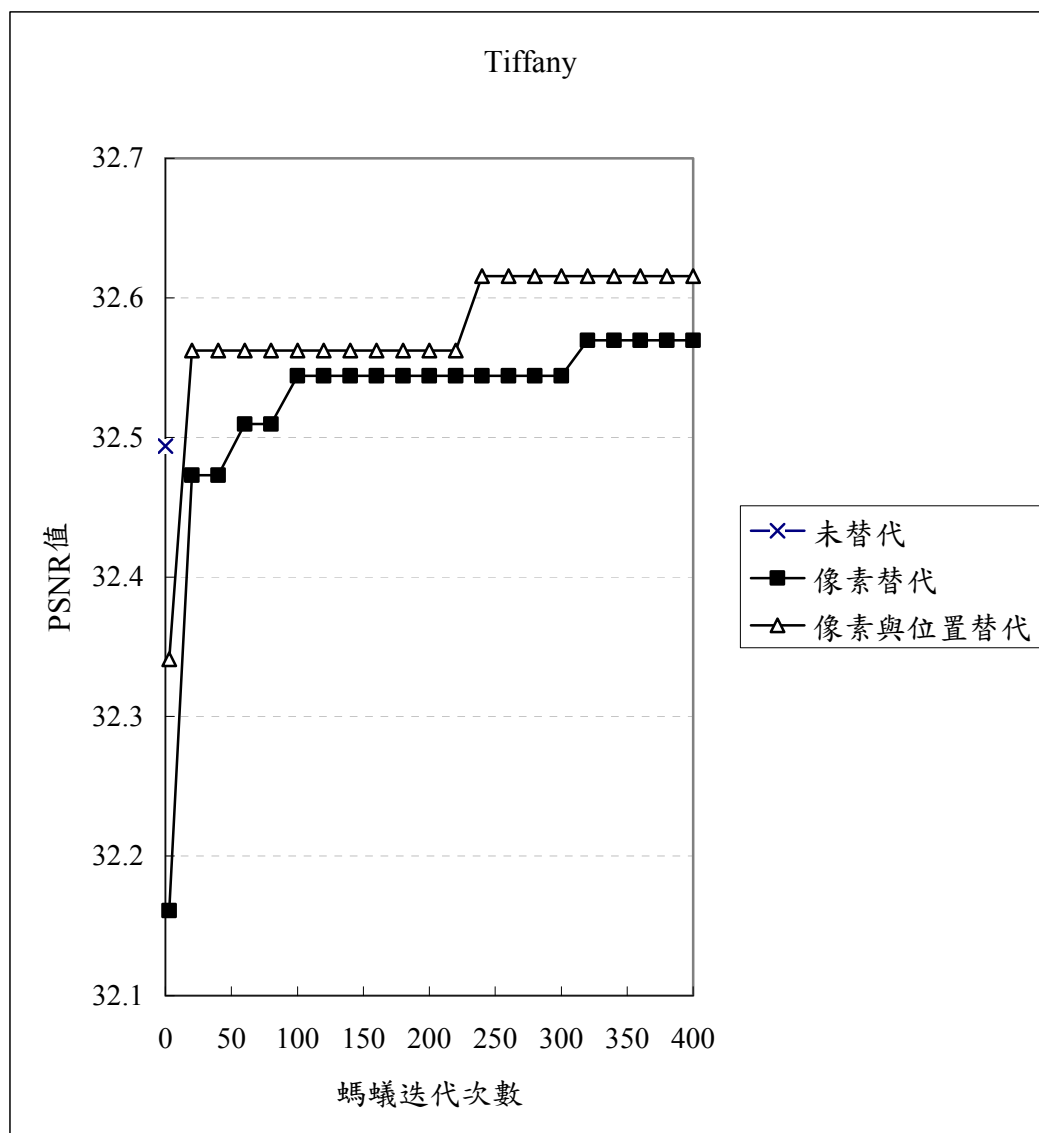


圖 54 Tiffany 螞蟻迭代次數與 PSNR 值分佈圖 (藏入 4 位元)

14. 掩護影像:Toys

表 61 Toys 影像介紹（藏 4 位元）

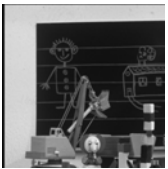

影像類別	掩護影像	秘密訊息
影像名稱	Toys	秘密訊息 1
影像大小	512×512	256×512
影像		

表 62 Toys 實驗數據（藏 4 位元）

方法	PSNR 值
無替代矩陣	32.688803
16×16 顏色替代矩陣	32.718443
16×16 顏色替代矩陣與 16×16 位置替代矩陣	32.767944

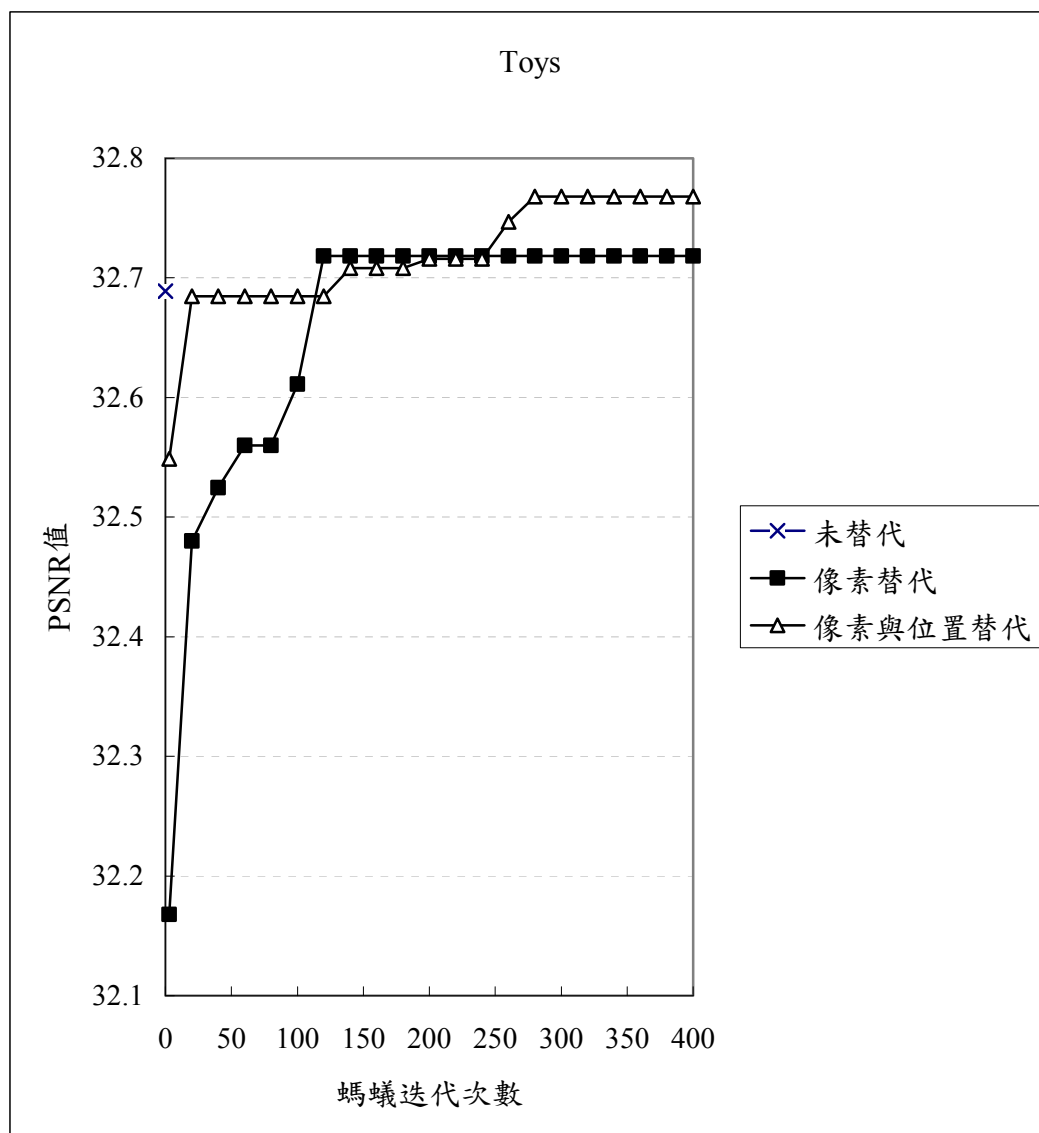


圖 55 Toys 螞蟻迭代次數與 PSNR 值分佈圖（藏入 4 位元）

15. 掩護影像:Zelda

表 63 Zelda 影像介紹 (藏 4 位元)


影像類別	掩護影像	秘密訊息
影像名稱	Zelda	秘密訊息 1
影像大小	512×512	256×512
影像		

表 64 Zelda 實驗數據 (藏 4 位元)

方法	PSNR 值
無替代矩陣	32.364788
16×16 顏色替代矩陣	32.419603
16×16 顏色替代矩陣與 16×16 位置替代矩陣	32.489155

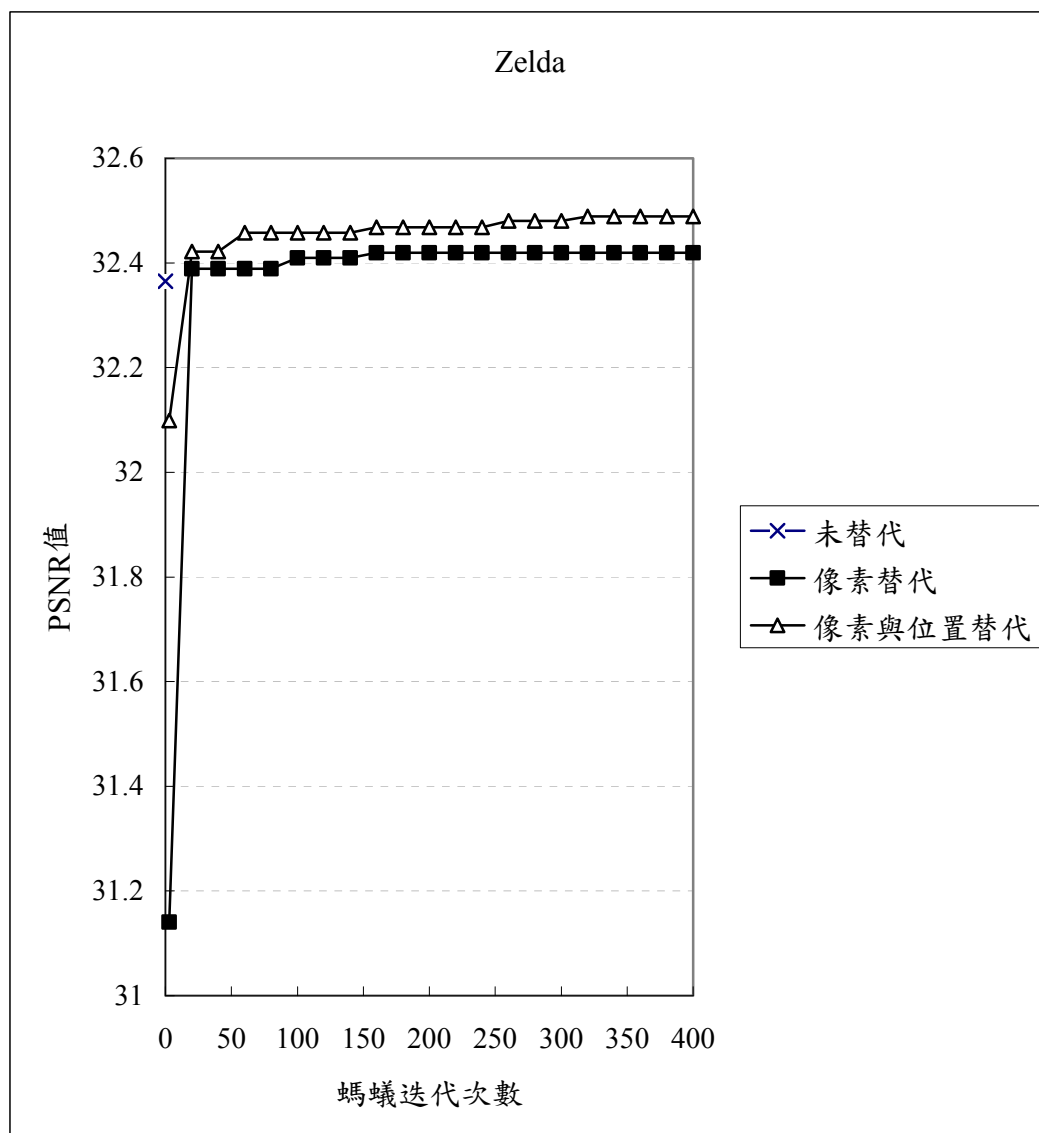


圖 56 Zelda 螞蟻迭代次數與 PSNR 值分佈圖 (藏入 4 位元)

16. 掩護影像:銘傳大學

表 65 銘傳大學影像介紹 (藏 4 位元)

影像類別	掩護影像	秘密訊息
影像名稱	銘傳大學	秘密訊息 1
影像大小	512×512	256×512
影像		

表 66 銘傳大學實驗數據 (藏 4 位元)

方法	PSNR 值
無替代矩陣	32.979894
16×16 顏色替代矩陣	33.003597
16×16 顏色替代矩陣與 16×16 位置替代矩陣	33.069747

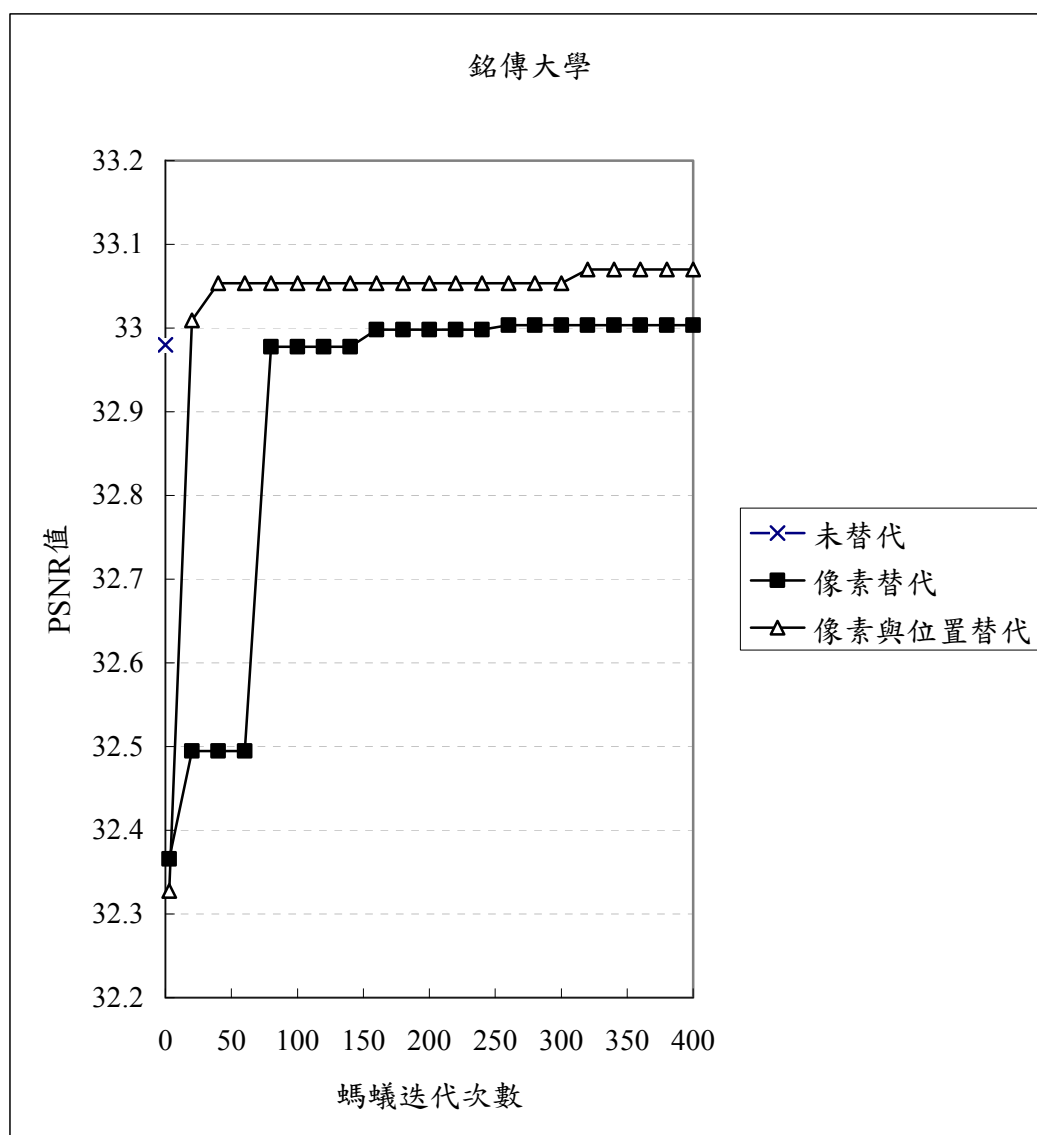


圖 57 銘傳大學螞蟻迭代次數與 PSNR 值分佈圖 (藏入 4 位元)

17. 掩護影像:銘傳資管

表 67 銘傳資管影像介紹 (藏 4 位元)

影像類別	掩護影像	秘密訊息
影像名稱	銘傳資管	秘密訊息 1
影像大小	512×512	256×512
影像		

表 68 銘傳資管實驗數據 (藏 4 位元)

方法	PSNR 值
無替代矩陣	32.364505
16×16 顏色替代矩陣	32.551805
16×16 顏色替代矩陣與 16×16 位置替代矩陣	32.563168

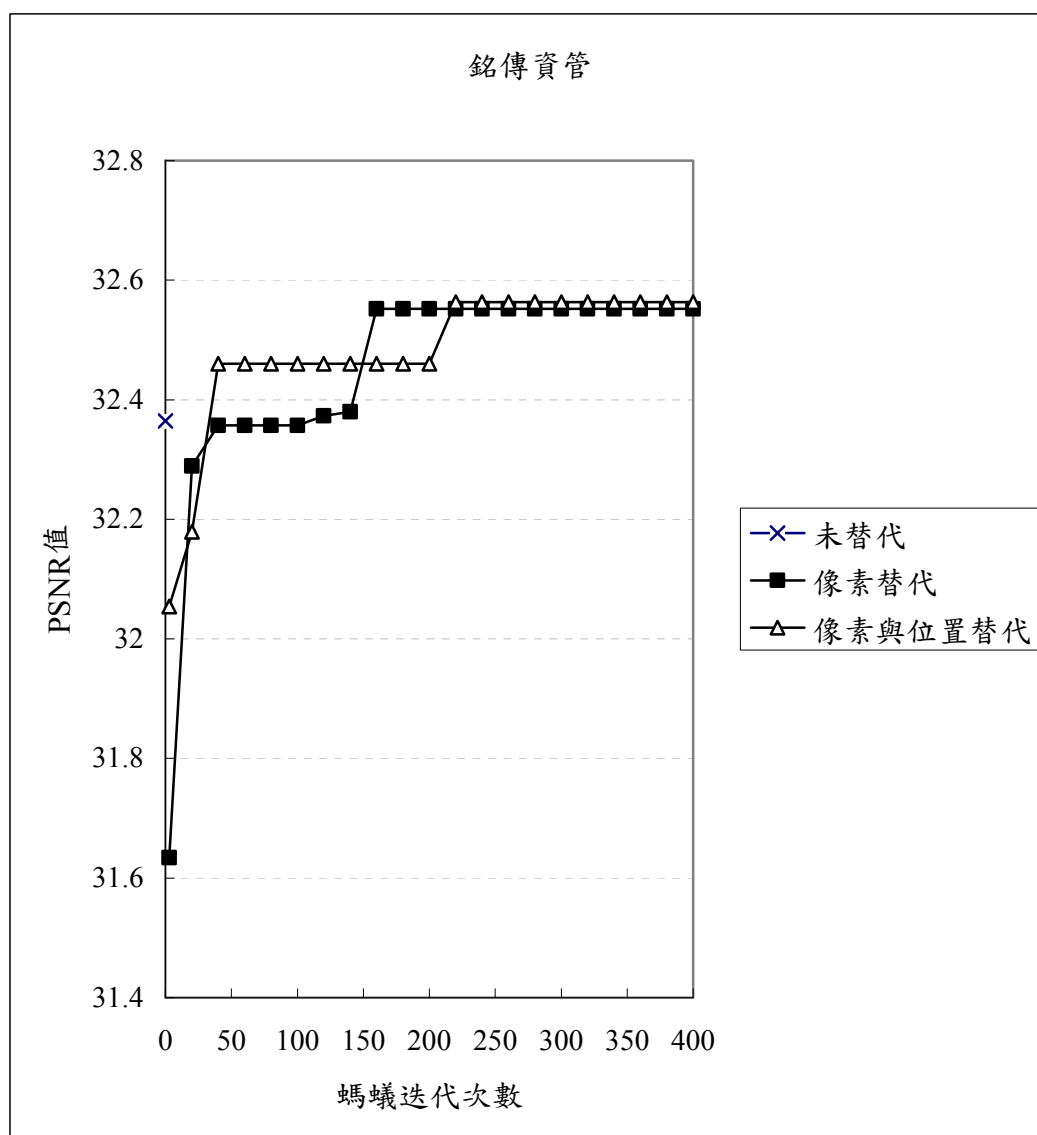


圖 58 銘傳資管螞蟻迭代次數與 PSNR 值分佈圖 (藏入 4 位元)

18. 掩護影像:Airplane

表 69 Airplane 影像介紹(藏 4 位元;秘密訊息 2)



影像類別	掩護影像	秘密訊息
影像名稱	Airplane	秘密訊息 2
影像大小	512×512	256×512
影像		

表 70 Airplane 實驗數據(藏 4 位元;秘密訊息 2)

方法	PSNR 值
無替代矩陣	32.579779
16×16 顏色替代矩陣	32.603425
16×16 顏色替代矩陣與 16×16 位置替代矩陣	32.686325

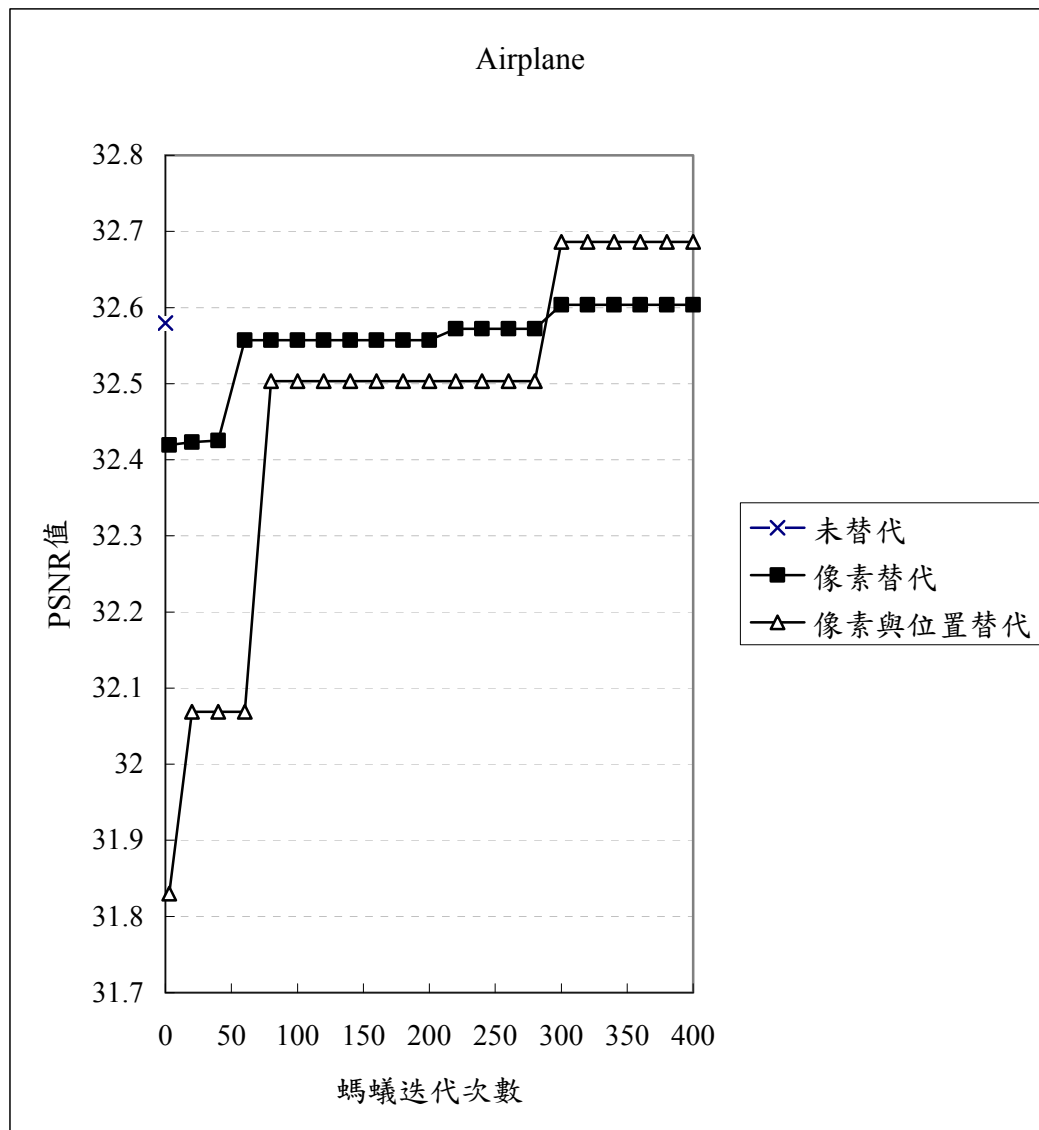


圖 59 Airplane 螞蟻迭代次數與 PSNR 值分佈圖 (藏 4 位元;秘密訊息 2)

19. 掩護影像: Baboon

表 71 Baboon 影像介紹 (藏 4 位元; 秘密訊息 2)



影像類別	掩護影像	秘密訊息
影像名稱	Baboon	秘密訊息 2
影像大小	512×512	256×512
影像		

表 72 Baboon 實驗數據 (藏 4 位元; 秘密訊息 2)

方法	PSNR 值
無替代矩陣	32.605081
16×16 顏色替代矩陣	32.627518
16×16 顏色替代矩陣與 16×16 位置替代矩陣	32.719601

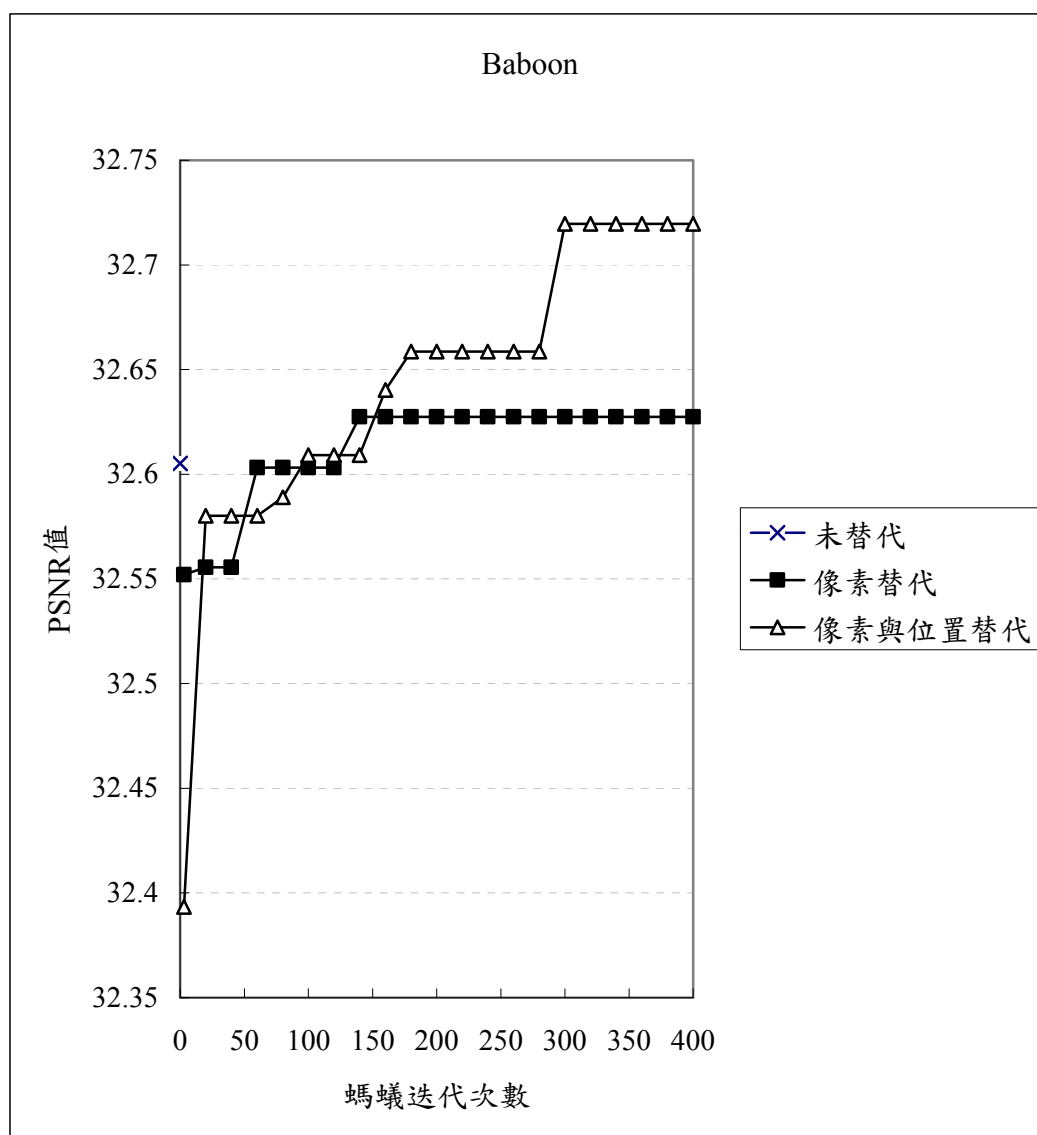


圖 60 Baboon 蟻蟻迭代次數與 PSNR 值分佈圖 (藏 4 位元; 秘密訊息 2)

20. 掩護影像: Bird1

表 73 Bird1 影像介紹 (藏 4 位元; 秘密訊息 2)

影像類別	掩護影像	秘密訊息
影像名稱	Bird1	秘密訊息 2
影像大小	512×512	256×512
影像		

表 74 Bird1 實驗數據 (藏 4 位元; 秘密訊息 2)

方法	PSNR 值
無替代矩陣	32.877180
16×16 顏色替代矩陣	32.990972
16×16 顏色替代矩陣與 16×16 位置替代矩陣	32.004074

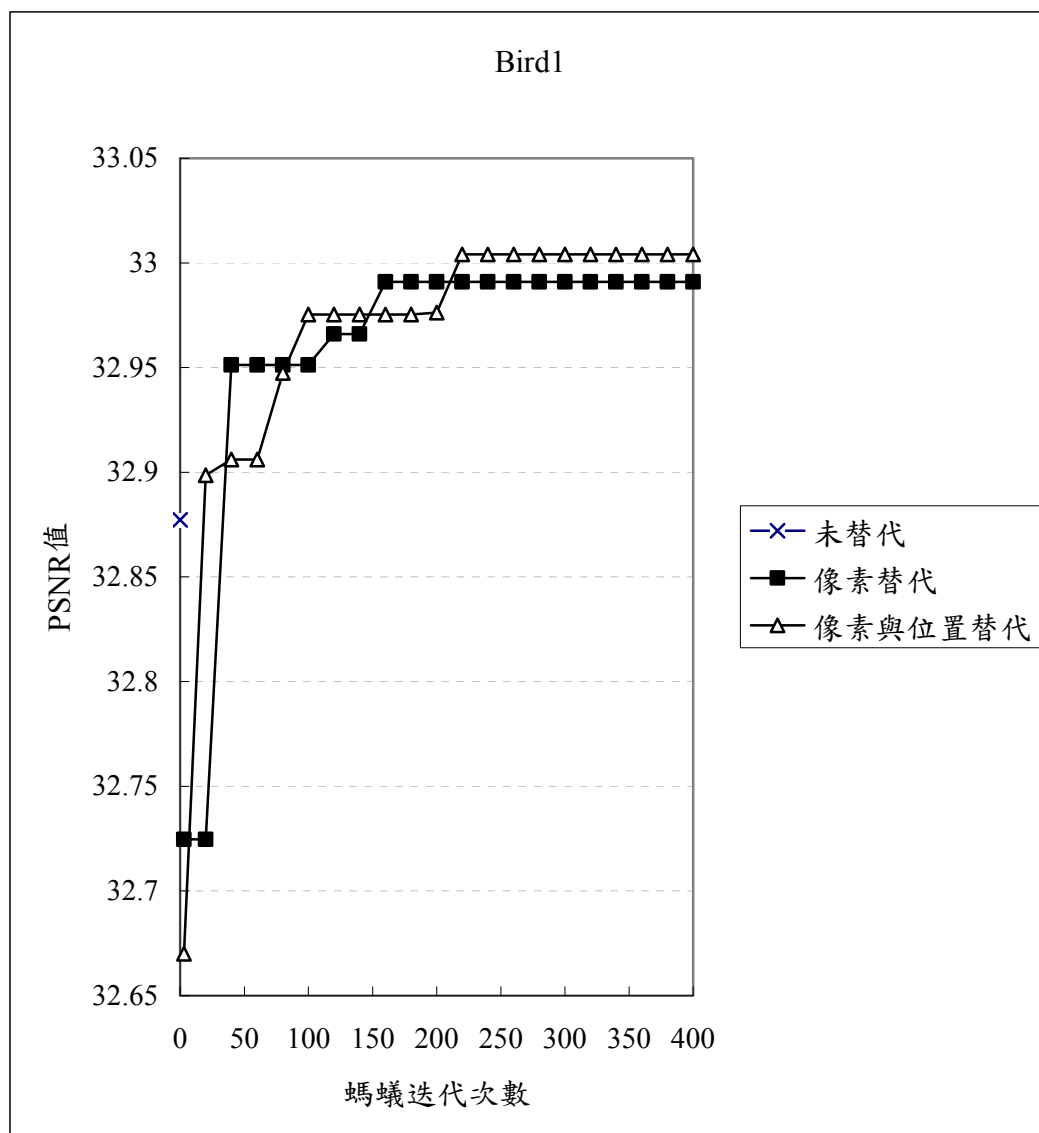


圖 61 Bird1 螞蟻迭代次數與 PSNR 值分佈圖 (藏入 4 位元; 秘密訊息 2)

21. 掩護影像:Boat

表 75 Boat 影像介紹 (藏 4 位元；秘密訊息 2)

影像類別	掩護影像	秘密訊息
影像名稱	Boat	秘密訊息 2
影像大小	512×512	256×512
影像		

表 76 Boat 實驗數據 (藏 4 位元；秘密訊息 2)

方法	PSNR 值
無替代矩陣	32.493757
16×16 顏色替代矩陣	32.530286
16×16 顏色替代矩陣與 16×16 位置替代矩陣	32.596687

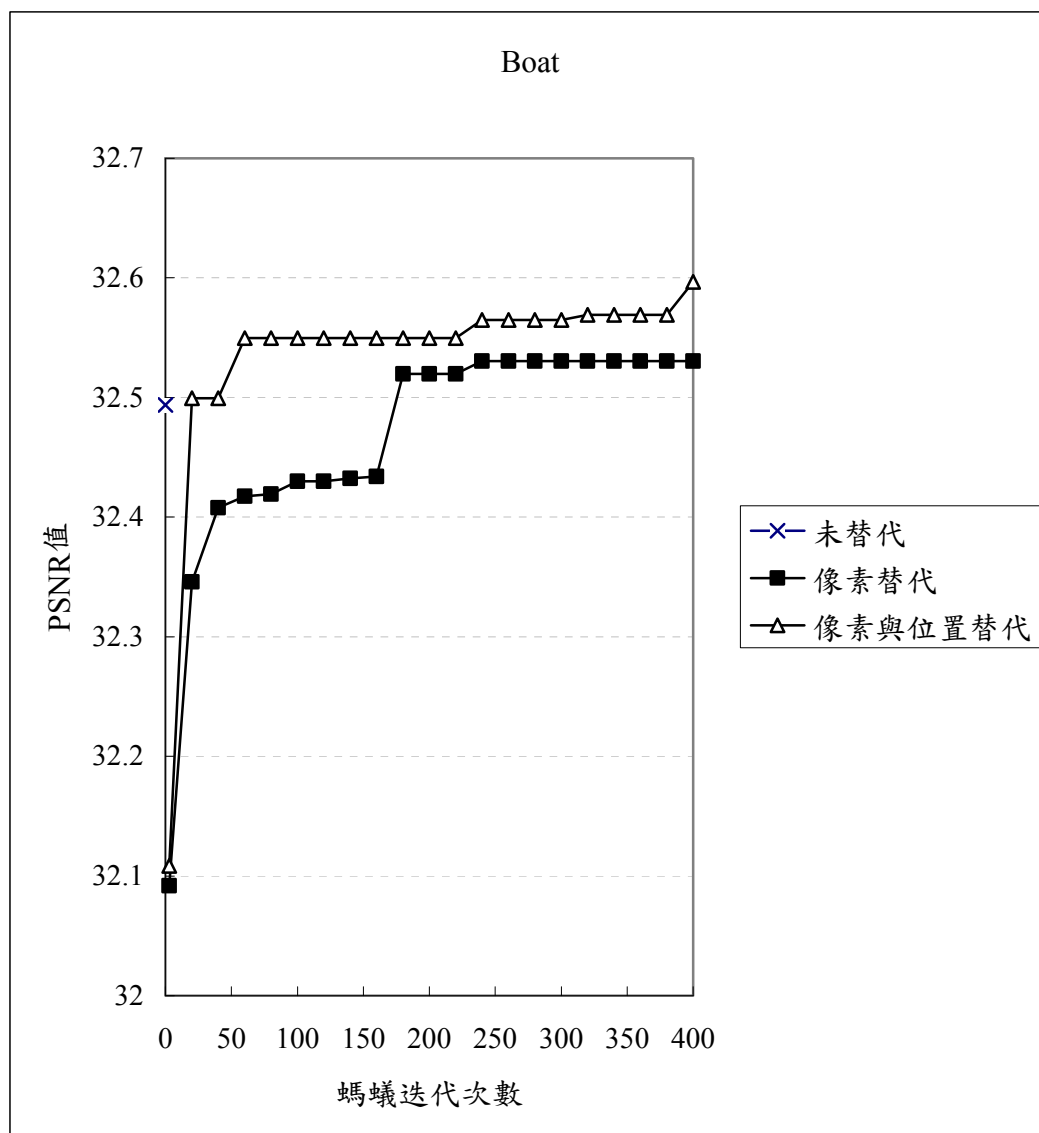


圖 62 Boat 螞蟻迭代次數與 PSNR 值分佈圖 (藏入 4 位元；秘密訊息 2)

22. 掩護影像:Girl

表 77 Girl 影像介紹 (藏 4 位元; 秘密訊息 2)



影像類別	掩護影像	秘密訊息
影像名稱	Girl	秘密訊息 2
影像大小	512×512	256×512
影像		

表 78 Girl 實驗數據 (藏 4 位元; 秘密訊息 2)

方法	PSNR 值
無替代矩陣	32.488575
16×16 顏色替代矩陣	32.523389
16×16 顏色替代矩陣與 16×16 位置替代矩陣	32.529968

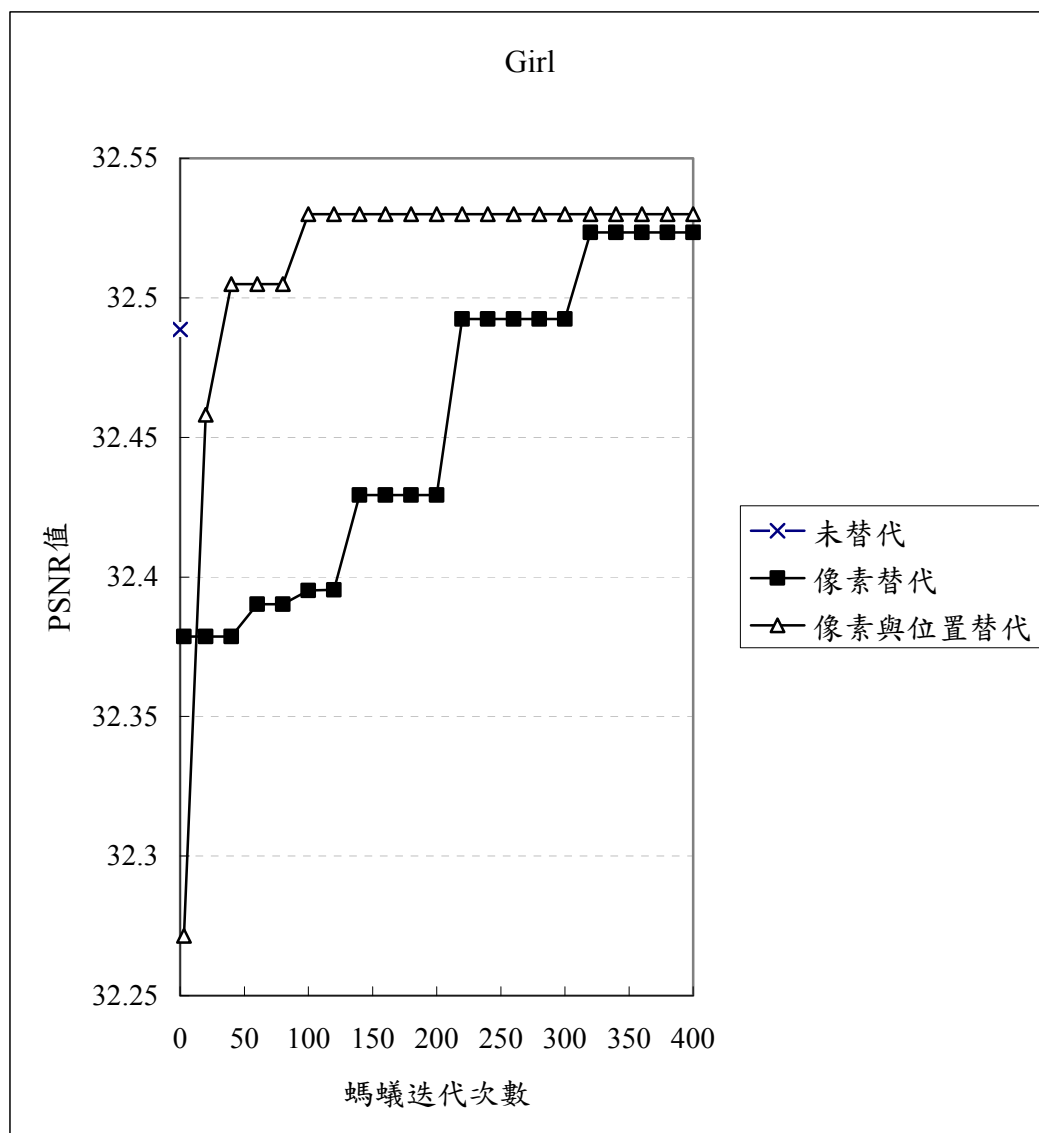


圖 63 Girl 螞蟻迭代次數與 PSNR 值分佈圖 (藏入 4 位元; 秘密訊息 2)

23. 掩護影像:Lenna

表 79 Lenna 影像介紹（藏 4 位元；秘密訊息 2）

影像類別	掩護影像	秘密訊息
影像名稱	Lenna	秘密訊息 2
影像大小	512×512	256×512
影像		

表 80 Lenna 實驗數據（藏 4 位元；秘密訊息 2）

方法	PSNR 值
無替代矩陣	32.576256
16×16 顏色替代矩陣	32.591210
16×16 顏色替代矩陣與 16×16 位置替代矩陣	32.672756

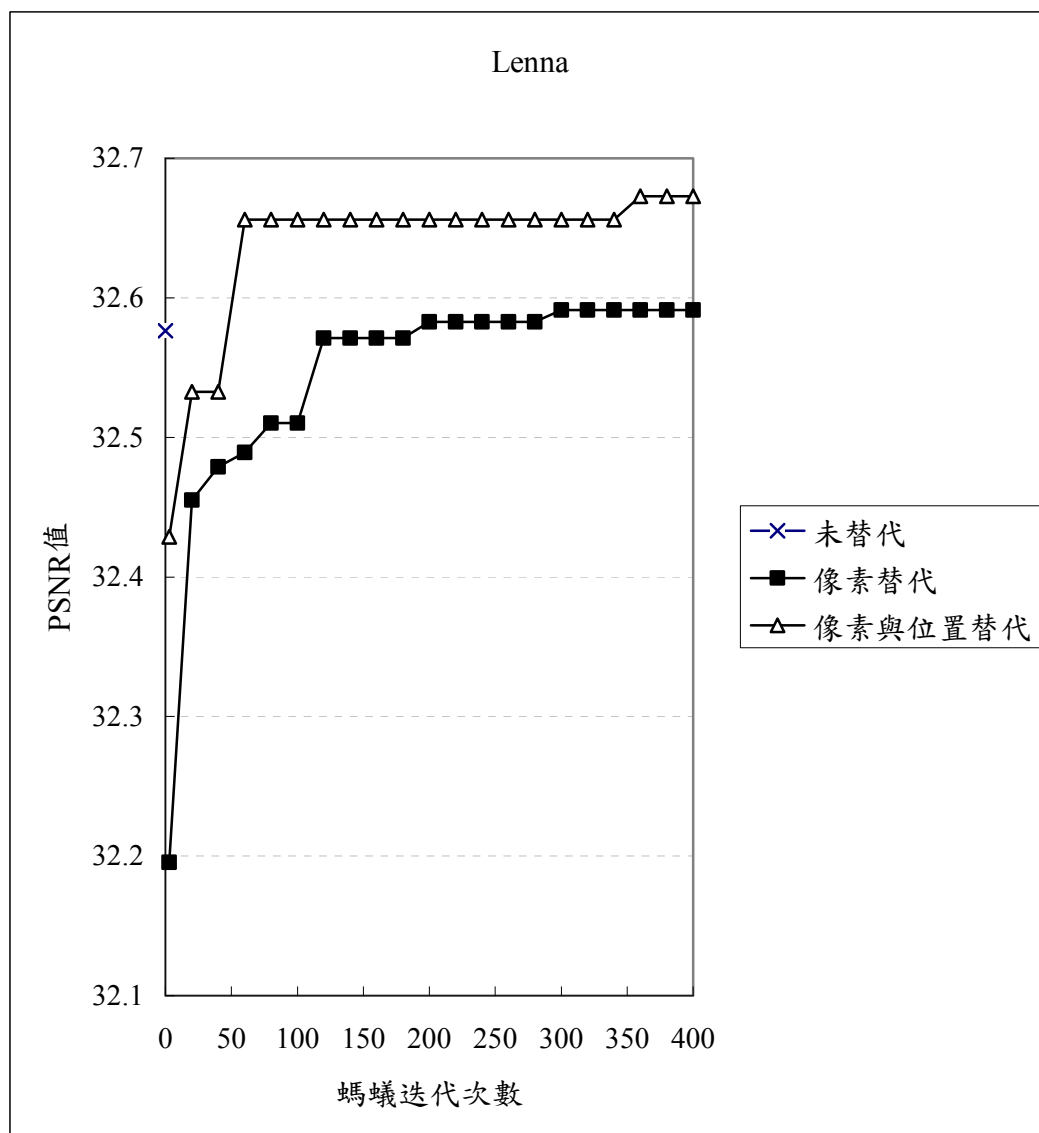


圖 64 Lenna 螞蟻迭代次數與 PSNR 值分佈圖（藏入 4 位元；秘密訊息 2）

24. 掩護影像:Pepper

表 81 Pepper 影像介紹 (藏 4 位元; 秘密訊息 2)



影像類別	掩護影像	秘密訊息
影像名稱	Pepper	秘密訊息 2
影像大小	512×512	256×512
影像		

表 82 Pepper 實驗數據 (藏 4 位元; 秘密訊息 2)

方法	PSNR 值
無替代矩陣	32.556638
16×16 顏色替代矩陣	32.557020
16×16 顏色替代矩陣與 16×16 位置替代矩陣	32.703556

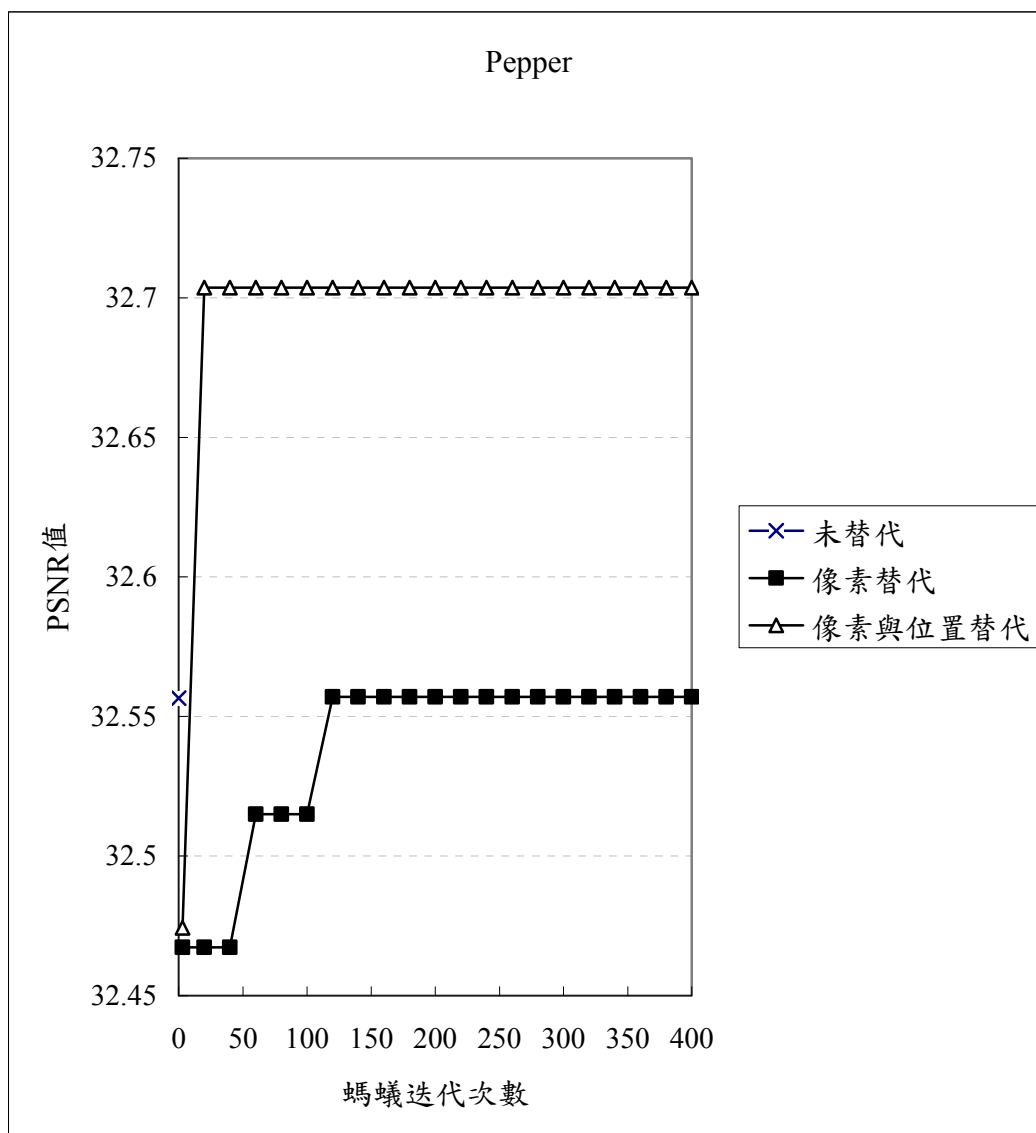


圖 65 Pepper 螞蟻迭代次數與 PSNR 值分佈圖 (藏入 4 位元; 秘密訊息 2)

4.4 實驗討論

在實驗中有許多重要的參數對於實驗的結果有重大的影響，其中最重要的莫過於費洛蒙蒸發係數 ρ 及隨機參數 q_0 ，這兩個因素是螞蟻是否會容易陷入局部最佳解的關鍵因素，其中，如果 ρ 很大，螞蟻就會很容易可以遺棄之前不好的路徑，但如果太大，螞蟻就完全無法根據費洛蒙去有效的尋找最短路徑，相反的，如果 ρ 很小，螞蟻就會很容易現在之前不好的路徑或是找到局部最佳解，甚至是無窮迴圈；而隨機參數 q_0 也是同樣的道理，當電腦產生隨機亂數小於 q_0 ，螞蟻的行走路徑則會被費洛蒙與經驗值來決定，大於 q_0 則是隨機選擇一條隨機路徑，不論 q_0 太大或太小都不好，都會讓螞蟻陷入局部最佳解或是無法根據費洛蒙找到好的解答。本研究最初設定 $\rho=0.1$ 、 $q_0=0.9$ 來進行實驗，但是卻常常無法突破找到更好的解，因此，之後便改變參數設定為 $\rho=0.1$ 、 $q_0=0.9$ 來進行實驗，結果類似，常常陷入局部最佳解。最後，改變參數設定為 $\rho=0.8$ 、 $q_0=0.3$ ，實驗結果與之前兩次結果相較起來，較不易陷入局部最佳解。根據 4.2 節及 4.3 節的實驗數據可以得知，螞蟻在參數設定為 $\rho=0.8$ 、 $q_0=0.3$ 時，不易陷入迴圈中。

在 4.2 節的實驗中，分別有四種不同的數據，第一種是沒有使用任何的替代矩陣的數據，第二種是只使用顏色替代矩陣的數據，第三種是使用具有相互關係的 4×4 的顏色替代矩陣及 4×4 的位置替代矩陣，第四種是使用具有相互關係的 4×4 顏色替代矩陣及 16×16 位置替代矩陣，我們以表 1、表 2 及圖 25 Airplane 這三個部份來作說明，當沒有使用任何替代矩陣時所得到的 PSNR 值為 44.375576，只使用 4×4 顏色替代矩陣所得到的 PSNR 值為 44.382295，而使用具有相互關係的 4×4 顏色替代矩陣與 4×4 位置替代矩陣所得到的 PSNR 值為 44.393490，由上述這三個數值可以知道：有使用替代矩陣來藏入秘密訊息的偽裝圖與沒有用替代矩陣來藏入秘密訊息的偽裝圖相比較其 PSNR 值會提高，而使用具有相互關係的 4×4 顏色替代矩陣與 4×4 位置替代矩陣，其 PSNR 值提高的更多，最後，使用具有相互關係的 4×4 顏色替代矩陣與 16×16 位置替代矩陣

所得到的 PSNR 值為 44.411240，比上述所提到的 PSNR 值都還高，由此可知，位置替代矩陣越大，與 PSNR 值呈正比。

而圖 26 Baboon、圖 28 Bird1、圖 30 Cat、圖 38 Toys 及圖 41 銘傳資管這五張圖的實驗數據中，有使用 4×4 的顏色替代矩陣與完全沒有使用何替代矩陣所得到的 PSNR 值是相同的，也就是說，在這五張圖的實驗中，顏色替代矩陣對於提升 PSNR 值完全沒有幫助，但是，在加入位置替代矩陣後所得到的 PSNR 值都有所提升，由此可知位置替代矩陣是非常重要的。

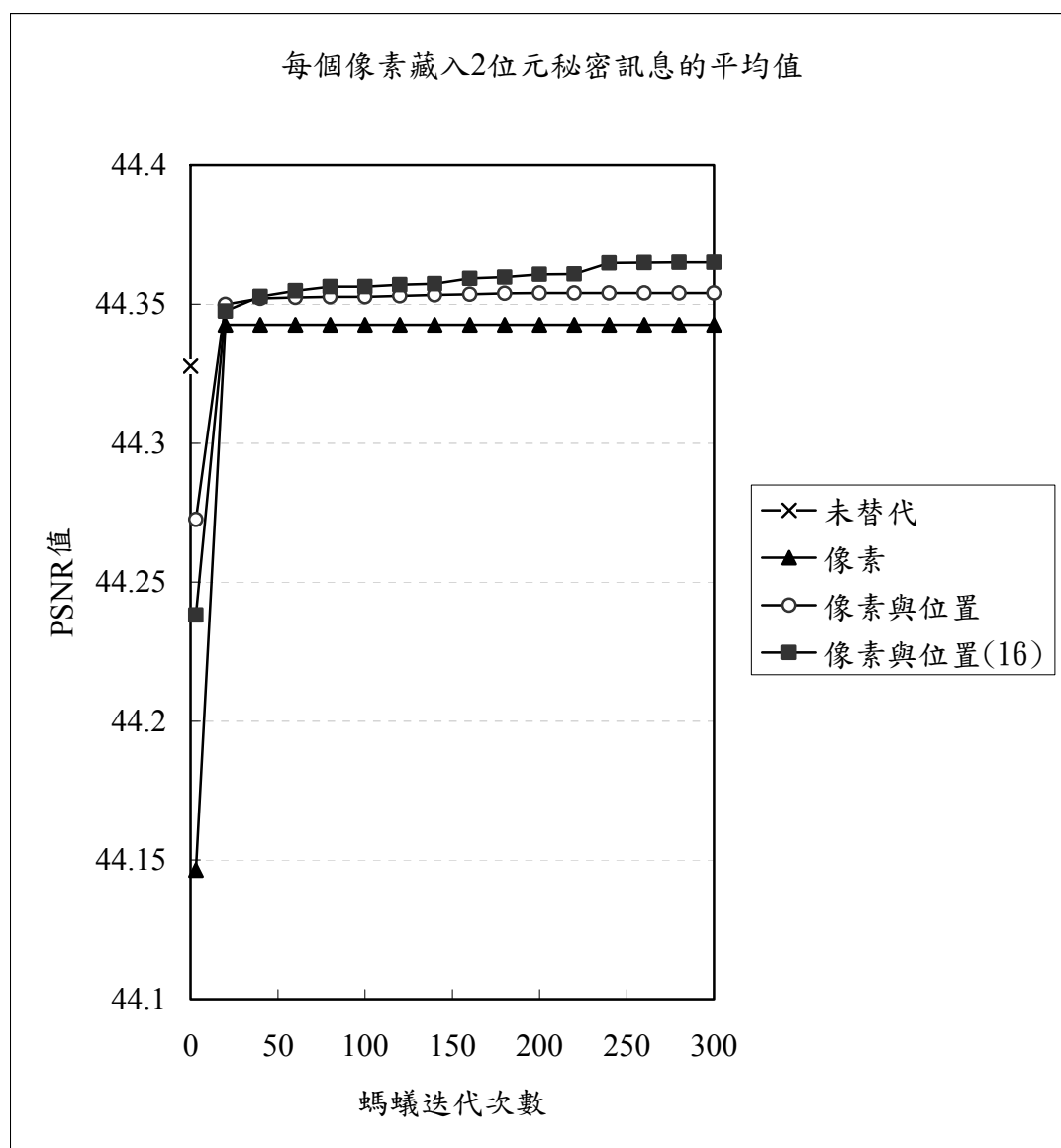


圖 66 4.2 節的實驗數據的平均值

圖 66 是 4.2 節中所有實驗數據的平均值，我們觀察螞蟻的迭代次數及 PSNR 值之間的關係可以發現到：第二種實驗很快的就找到最佳解，其次是第三種實驗，大約迭代次數 100 次上下就找到最佳解了，而第四種實驗，雖然起初的 PSNR 值不高，或許是這四種實驗當中最差的，但是經過 300 次的迭代次數後，其 PSNR 值總能夠大於其他三種實驗，而且如果再繼續進行實驗，其 PSNR 值似乎還能夠更高。這是因為第四種實驗的位置替代矩陣為 16×16 的矩陣，其解答空間較其他三種實驗大許多。所以在經過 300 次的迭代後，其 PSNR 值還很有成長的空間。

在 4.3 節的實驗中，分別有三種不同的實驗數據，第一種是沒有使用任何的替代矩陣的數據，第二種是只使用 16×16 的顏色替代矩陣的數據，第三種是使用具有相互關係的 16×16 的顏色替代矩陣及 16×16 的位置替代矩陣，我們以表 49、表 50 及圖 49 這三個部份作說明，當沒有使用任何替代矩陣時所得到的 PSNR 值為 32.420765，只使用 16×16 顏色替代矩陣所得到的 PSNR 值 32.464535，而使用具有相互關係的 16×16 顏色替代矩陣與 16×16 位置替代矩陣所得到的 PSNR 值為 32.546227，由上述三個數值可以清楚的知道：運用顏色替代矩陣可以改善偽裝影像被破壞的程度，而同時使用具有相互關係的 16×16 顏色替代矩陣與 16×16 位置替代矩陣更加可以改善偽裝影像被破壞的程度。

而在 4.3 節實驗中，有許多張圖進行了藏入 2 種不同的秘密訊息的實驗，我們來比較實驗的結果，以 Baboon 這張圖來說明，其中表 37、表 38 及圖 43 說明了 Baboon 藏入了第一種秘密訊息的實驗結果，而表 71、表 72 及圖 60 說明了 Baboon 藏入了第二種秘密訊息的實驗數據結果，我們由實驗結果可以得知：Baboon 這張掩護影像被藏入了第一種秘密訊息後，得到的 PSNR 值為 32.406961，而被藏入第二種秘密訊息後得到的 PSNR 值為 32.605081，會造成 PSNR 值不同的原因在於這兩種秘密訊息的影像特徵不同，其中，第一種秘密訊息的影像特徵與 Baboon 的影像特徵差異較大，所以其 PSNR 值較低，也就是藏

密秘密訊息後，其偽裝影像的影像品質較差。另外，Baboon 這張掩護影像運用 16×16 顏色替代矩陣藏入第一種秘密訊息後，得到的 PSNR 值為 32.520500，且藏入第二種秘密訊息後，得到的 PSNR 值為 32.627518，而運用具有相互關係的 16×16 顏色替代矩陣與 16×16 位置替代矩陣所得到的 PSNR 值為 32.546455，且藏入第二種秘密訊息後，得到的 PSNR 值為 32.719601，由上述的數值可以得知：有使用 16×16 顏色替代矩陣與沒有使用任何的替代矩陣來藏入第一種秘密訊息其 PSNR 值的差異為 0.113539，且有使用 16×16 顏色替代矩陣與沒有使用任何的替代矩陣來藏入第二種秘密訊息其 PSNR 值的差異為 0.022437；而使用具有相互關係的 16×16 顏色替代矩陣與 16×16 位置替代矩陣與只使用 16×16 顏色替代矩陣來藏入第一種秘密訊息，其 PSNR 值的差異為 0.025955，且使用具有相互關係的 16×16 顏色替代矩陣與 16×16 位置替代矩陣與只使用 16×16 顏色替代矩陣來藏入第二種秘密訊息，其 PSNR 值的差異為 0.092083。而造成上述的差異主要原因在於：第一種秘密訊息與第二種秘密訊息的影像特徵不同。由上述數值來分析，我們可以知道，Baboon 偽裝影像在藏入第一種秘密訊息時，光使用顏色替代矩陣就能夠對 PSNR 值產生很的影響，但是 Baboon 偽裝影像在藏入第二種秘密訊息時，只單純使用顏色替代矩陣使明顯不足夠的，而是必需使用具有相互關係的 16×16 顏色替代矩陣與 16×16 位置替代矩陣才能夠有效的改善偽裝影像被破壞的程度。

圖 67 是 4.3 節中藏入第一種秘密訊息的實驗數據的平均值，分別有三種不同的實驗，第一種是沒有使用任何的矩陣，第二種是使用 16×16 顏色替代矩陣，第三種是使用具有相互關係的 16×16 顏色替代矩陣與 16×16 位置替代矩陣，我們可以由圖 67 得知，螞蟻迭代次數在 10 次以內的時候，其 PSNR 值最高者為沒有使用任何替代矩陣的實驗，雖然第二種和第三種實驗剛開始的 PSNR 值都低於第一種實驗，但是隨著螞蟻迭代次數的增加，第二種實驗及第三種實驗的 PSNR 值很快的都會大於第一種實驗的 PSNR 值，而且第三種實驗的 PSNR 值增

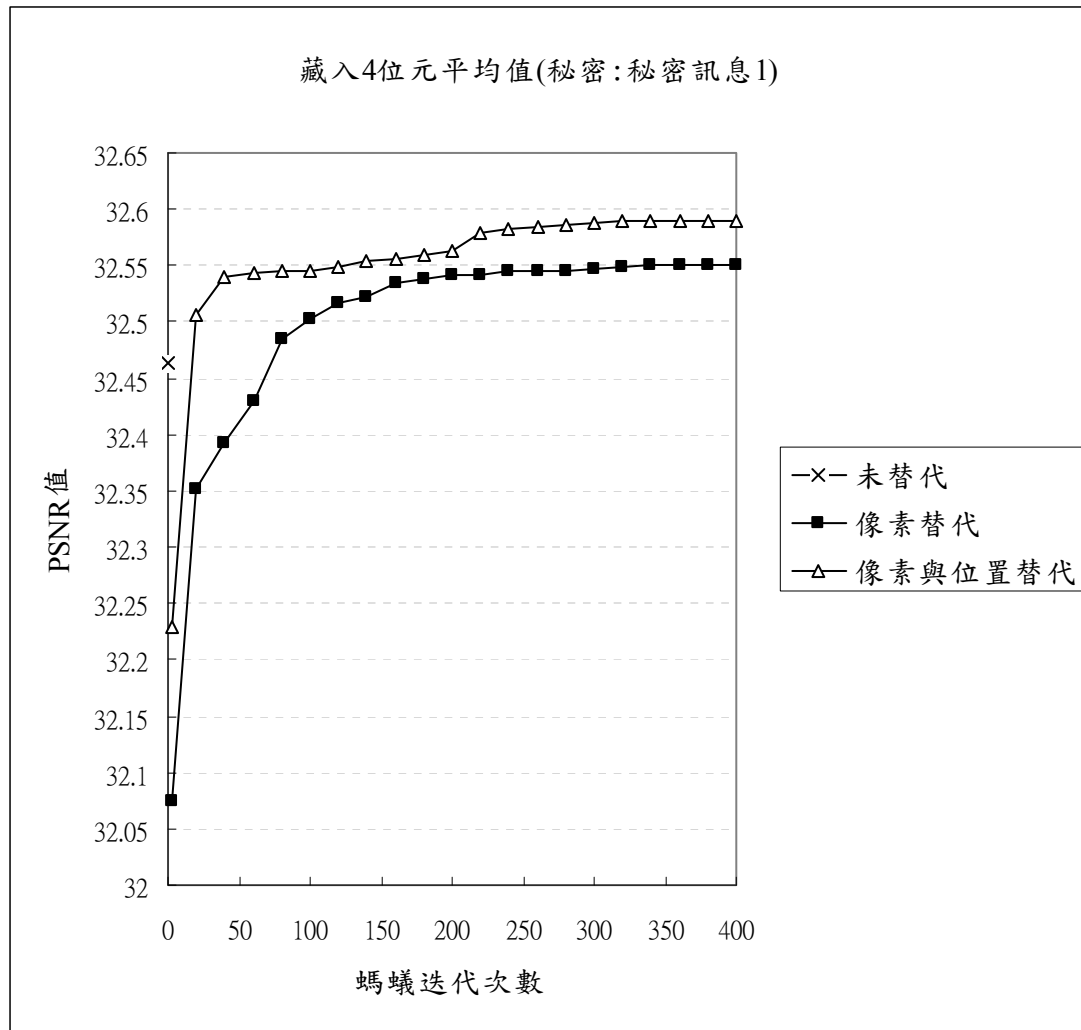


圖 67.4.3 節藏入第一種秘密訊息的實驗數據的平均值

加的速率較快，很快的就超越第一種實驗的 PSNR 值。我們比較第二種實驗與第三種實驗的曲線，可以看到螞蟻的迭代次數大約在 200 的時候，兩條曲線較相近，這或許是因為第三種實驗在螞蟻迭代次數 200 次時較易陷入局部最佳解所造成。

圖 68 是 4.3 節藏入第二種秘密訊息的實驗數據的平均值，分別有三種不同的實驗，第一種是沒有使用任何的矩陣，第二種是使用 16×16 顏色替代矩陣，第三種是使用具有相互關係的 16×16 顏色替代矩陣與 16×16 位置替代矩陣，我們可以由圖 67 得知，螞蟻迭代次數在 10 次以內的時候，其 PSNR 值最高者為沒有使用任何替代矩陣的實驗，雖然第二種和第三種實驗剛開始的 PSNR 值都

低於第一種實驗，且第三種實驗的 PSNR 值是最底的，但是隨著螞蟻迭代次數的增加，第二種實驗及第三種實驗的 PSNR 值都會大於第一種實驗的 PSNR 值，而且第三種實驗的 PSNR 值很快就超越另外兩種實驗的 PSNR 值。造成上述結果的主要原因是：第三種實驗中螞蟻的可能路徑組合非常多，導致螞蟻迭代次數 10 以下的時候，暫時還沒找到較好的路徑，但是隨著螞蟻迭代次數的增加，根據費洛蒙濃度及經驗的累積，很快的就能夠找到更好的路徑了，使得其 PSNR 值增加的速度很快，在螞蟻迭代次數 20 次的時候就超越第二種實驗的 PSNR 值，且在螞蟻迭代次數 100 次時就超越了第一種實驗的 PSNR 值。最後，根據曲線的趨勢來看，第二種實驗及第三種實驗的 PSNR 值還正在提升當中，且第三種實驗的增加速率仍舊高於第二種實驗。

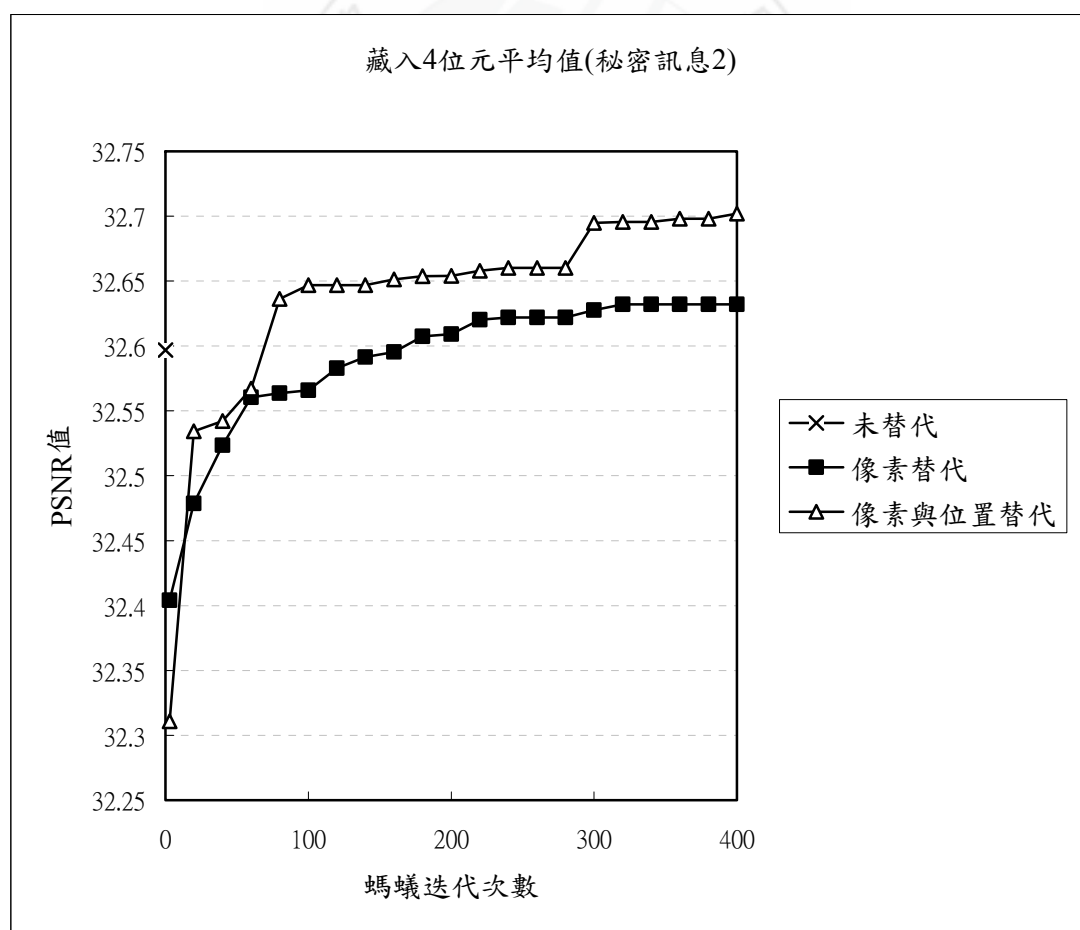


圖 68.4.3 節藏入第二種秘密訊息的實驗數據的平均值

最後根據 4.2 節及 4.3 節中的實驗數據，我們可以清楚的知道下列各點：

1. 有使用替代矩陣比沒有使用替代矩陣的 PSNR 值高或相同。
2. 有共同使用像素及位置替代矩陣的 PSNR 值比只使用顏色替代矩陣的 PSNR 值高或相同。
3. 位置替代矩陣越大，則更可以改善偽裝影像的影像品質被破壞的程度。
4. 儘管掩護影像是相同的，但是秘密訊息不同時，其偽裝影像被破壞的程度仍然不同，會受到影像特徵所影響。



第五章 結論

隨著時代的進步，資訊技術快速的蓬勃發展，資訊隱藏已成為很重要的一個議題，人們能夠藉由看似平常的傳遞文件或圖片，就能將秘密安全的傳達到目的地，並且不會被惡意者竊取，或被隨意的流傳。在本論文中，我們探討了許多資訊隱藏的方法，像是中古世紀的秘密隱藏法、在中國清朝時代的嵌字詩或二次大戰的用隱形墨水來書寫秘密訊息，甚至是現代的最低位元置換法，目的都是為了能增加機密資料的安全性。

其中，最低位元置換法被分成了許多種，不論是哪一種都有它的優缺點，像是簡單最低位元置換法，藏匿方式非常簡單方便，但最大的缺點就是秘密訊息的安全性不佳，容易被惡意者發覺或竊取；而最佳最低位元置換法改善了最低位元置換法的缺點，像是使用了顏色替代矩陣改變秘密訊息的像素值，能降低了偽裝影像被破壞的程度，並且先使用加密演算法打亂秘密訊息的位置，就算秘密訊息被惡意者取出了，仍舊無法知道真實的秘密訊息為何，這樣就能提升秘密訊息的安全性了，但是最佳最低位元置換法也有一個很大的缺點，就是在於求解顏色替代矩陣上會花費大量的時間，因此，之後又提出了基因演算法來求解顏色替代矩陣來解決這個問題。

而本論文因基因演算法的效率及答案的品質不佳，因此應用了蟻群最佳化演算法來改善這個問題，因為蟻群最佳化演算法不需要將問題轉換為二元化編碼，所以沒有二元編碼基因演算法所產生的問題，不會浪費時間在編碼轉換上，而且蟻群最佳化演算法可以利用費洛蒙參數及啟發值使螞蟻能夠快速的找到好的路徑，同時也可避免陷入局部最佳解，這是基因演算法所沒有的，再加上蟻群最佳化演算法有探索新路徑的功能，相當於基因演算法中的「突變」，這使螞蟻能夠找尋到新的路徑，也可避免陷入局部最佳解，來快速提升偽裝圖的 PSNR 值，增加偽裝圖的安全性。

再者，本論文認為用顏色替代矩陣來改善偽裝影像被破壞的程度的能力有

限，因此，本論文提出了位置替代矩陣，並結合顏色替代矩陣及位置替代矩陣來共同改善偽裝影像被破壞的程度。最後，根據實驗結果顯示，使用蟻群最佳化演算法所求解出的顏色替代矩陣及位置替代矩陣都能有效的改善偽裝影像被破壞的程度，其中同時使用顏色替代矩陣及位置替代矩陣的改善程度都優於只使用顏色替代矩陣的改善程度，其偽裝圖的 PSNR 值皆優於只使用顏色替代矩陣的偽裝圖的 PSNR 值，甚至，如果位置替代矩陣越大，秘密訊息被分割的區塊越多，越能改善偽裝影像的被破壞的程度，以更能夠提升秘密訊息的安全性。



參考文獻

1. 王旭正，柯建萱(民96)。資訊媒體安全-偽裝學與數位浮水印。台北：博碩文化股份有限公司。
2. 黃義美，黃仁俊(民90)。資訊隱蔽技術。資訊與教育雜誌，第83期，頁46-51。
3. Ahuja, R., Orlin, J.B., and Tivari, A. (2000). A greedy genetic algorithm for the quadratic assignment problem. *Computers, Operations Research*, 27(10), 917-934.
4. Aslantas, V., Ozer, S., and Ozturk, S. (2008). A novel fragile watermarking based on particle swarm optimization. *Multimedia and Expo, 2008 IEEE International Conference on*, 269-272.
5. Bender, W., Gruhl, D., Morimoto, N., and Lu, A. (1995). Techniques for data hiding. *IBM Systems Journal*, 35(3-4), 313-336.
6. Chan, C.K. and Cheng, L.M. (2004). Hiding data in images by simple LSB substitution. *Pattern Recognition*, 37(3), 469-474.
7. Chang, C.C., Chen, T.S., and Chung, L.Z. (2002). A steganographic method based upon JPEG and quantization table modification. *Information Science*, 141, 123-138.
8. Chang, C.C., Lin, M.H., and Hu, Y.C. (2002). A fast and secure image hiding scheme based on LSB substitution. *Pattern Recognition and Artificial Intelligence*, 16(4), 399-416.
9. Chen, T.H. and Tsai, D.S. (2006). Owner-customer right protection mechanism using a watermarking scheme and a watermarking protocol. *Pattern Recognition*, 39(8), 1530-1541.
10. Chun, W.D. and Hsiang, T.E. (2002). Data hiding in images via multiple-based number conversion and lossy compression. *Consumer Electronics, IEEE Transactions on*, 44(4), 1406-1412.
11. Das, T.K. and Maitra, S. (2006). Analysis of the wavelet tree quantization watermarking strategy and a modified robust scheme. *Multimedia Systems*, 12(2), 151-163.
12. Dorigo, M., Maniezzo, V., and Colorni, A. (1996). Ant system: optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Man, and Cybernetics-part B: Cybernetics*, 26(1), 29-41.
13. Dorigo, M., and Thomas, S. (2004). *Ant Colony Optimization*. Cambridge: the MIT Press, 1-120.
14. Holland, J.H., *Adaptive in Natural and Artificial Systems*, Ann Arbor, MI: Univ. Michigan Press, 1975.
15. Hou, Y.C. and Chen, P.M. (2000). An asymmetric watermarking scheme based on

- visual cryptography. *Proceedings of the 5th International Conference on Signal Processing*, 992-995.
16. Huang, F., Guan, Z., and Wu, X. (2006). Blind watermarking algorithm based on DCT. *Journal of Huazhong University of Science and Technology*, 34(2), 17-19.
 17. Katzenbeisser, S. and Petitcolas, F.A.P. (2000). *Information Hiding Techniques for Steganography and Digital Watermarking*. Boston: Artech House, 45-47.
 18. Kennedy, J. and Eberhart, R.C. (1995). Particle swarm optimization. *Proceedings of IEEE International Conference on Neural Networks*, 1942-1948.
 19. Lei, C.L., Yu, P.L., Tsai, P.L., and Chan, M.H. (2004). An efficient and anonymous buyer-seller watermarking protocol. *IEEE Transactions on Image Processing*, 13(12), 1618-1626.
 20. Li, X. and Wang, J. (2007). A steganographic method based upon JPEG and particle swarm optimization algorithm. *Information Sciences*, 177, 3099-3109.
 21. Lin, C.C., and Tsai, W.H. (2003). Visual cryptography for gray-level images by dithering techniques. *Pattern Recognition Letters*, 24, 349-358.
 22. Linde, Y., Buzo, A., and Gray, R.M. (1980). An algorithm for vector quantizer design. *IEEE transactions on communications*, 28, 84-95.
 23. Lou, D.C., Tso, H.K., and Liu, J.L. (2007). A copyright protection scheme for digital image using visual cryptography technique. *Computer Standards & Interfaces*, 29, 125-131.
 24. Memon, N. and Wong, P.W. (2001). A buyer-seller watermarking protocol. *IEEE Transactions on Image Processing*, 10(4), 643-649.
 25. Merkle, D., Middendorf, M., and Schmeck, H. (2002). Ant colony optimization for resource-constrained project scheduling. *IEEE Transactions on Evolutionary Computation*, 6(4), 333-346.
 26. Naor, M. and Shamir, A. (1995). Visual cryptography. in *Advances in Cryptography-EUROCRYPT'94*, 1-12.
 27. Noda, H., Niimi, M., and Kawaguchi, E. (2006). High-performance JPEG steganography using quantization index modulation in DCT domain. *Pattern Recognition Letters*, 27(5), 455-461.
 28. Petitcolas, F.A.P., Anderson, R.J., and Kuhn, M.G. (1999). Information hiding survey. *Proceedings of the IEEE*, 87(7), 1062-1078.
 29. Qiao, L. and Nahrstedt, K. (1998). Watermarking schemes and protocols for protecting rightful ownership and customer's rights. *Journal of Visual Communication and Image Representation*, 9(3), 194-210.
 30. Thin, C.C. and Lin, J.C. (2003). A simple and high-hiding capacity method for hiding digit-by-digit data in image based on modulus function. *Pattern Recognition*, 36, 2875-2881.

31. Tsai, M.J. and Hung, H.Y. (2004). DCT and DWT-based image watermarking by using subsampling. *Proceedings of the 24th International Conference on Distributed Computing Systems Workshops*, 184-189.
32. Upham, D., Jpeg-Jsteg-v4.
<http://www.funte.fi/pub/crypt/steganography/Jpeg-v4.diff.gz>.
33. Voyatzis, G. and Pitas, I. (1996). Applications of toral automorphisms in image watermarking. *Proceedings of International Conference on Image Processing*, 2, 237-240.
34. Wang, R.Z., Lin, C.F., and Lin, J.C. (2001). Image hiding by optimal LSB substitution and genetic algorithm. *Pattern Recognition*, 34(3), 671-683.
35. Wang, Z., Shi, B., and Wang, N. (2007). Robust watermarking algorithm using wavelet transform and HVS. *Journal of Huazhong University of Science and Technology*, 35(1), 26-28.
36. Wang, F.H., Yen, K.K., Jain, L.C., and Pan, J.S. (2007). Multiuser-based shadow watermark extraction system. *Information Sciences*, 177(21), 2522-2532.
37. Whitely, D.A. A genetic algorithm tutorial. *Statistics and Computing* 1994, 4, 65-85.
38. Zhang, J. and Cui, L. (2006). New robust digital watermark technique based on wavelet transform. *Journal of Information and Computational Science*, 3(1), 137-142.
39. Zhang, J., Kou, W., and Fan K. (2006). Secure buyer-seller watermarking protocol. *IEE Proceedings: Information Security*, 153(1), 15-18.
40. Zhang, X.P., and Wang, S.Z. (2005). Steganography using multiple-base notational system and human vision sensitivity. *IEEE Signal Processing Letters*, 12(1), 67-70.
41. Zhou, C.H., Kou, J.S., and Li, M.Q. (2007). Digital image watermark detector design based on a distribution in DCT domain. *Journal of Optoelectronics Laser*, 18(7), 838-841.

著作

1. 王尹良、劉妍芝、許慶昇 (2009, 06, 3-5)。結合機率理論的數位影像竄改偵測方法，第十九屆資訊安全會議論文集光碟(CISC 2009)，台北市。(學生最佳論文競賽佳作獎)
2. 劉妍芝、王尹良、許慶昇 (2009, 06, 3-5)。以蟻群最佳化演算法為基礎的最低位元置換法，第十九屆資訊安全會議論文集光碟(CISC 2009)，台北市。
3. 許慶昇、王尹良、劉妍芝(2009, 05, 23)。機率式兩階段數位影像驗證方法，第二十屆國際資訊管理學術研討會論文集光碟(ICIM 2009)，台北市。
4. 許慶昇、劉妍芝、王尹良(2009, 05, 23)。結合蟻群最佳化演算法與改良式替代矩陣之最低位元置換法，第二十屆國際資訊管理學術研討會論文集光碟(ICIM 2009)，台北市。

