漸進式機密三維模型分享演算法

林彥宏 國立中興大學 資訊科學系大學生
陳沛敬 國立中興大學 資訊科學系大學生
柏傑 國立中興大學 資訊科學系大學生
陶嘉瑋 國立中興大學 資訊科學系大學生
吳國禎 國立中興大學 資訊科學系研究生
王宗銘 國立中興大學 資訊科學系教授
e-mail：cmwang@cs.nchu.edu.tw

摘要

三維模型已大量應用在電影與電玩遊戲，此也使得在網路上機密分享並漸進擷取三維模型成為一個重要的研究課題。目前的三維模型機密分享演算法均無法漸進式的還原三維模型。此外，若強將二維影像的漸進式演算法使用在三維模型將導致還原出的模型具有極大的失真率，且成圖之圖像品質也不佳。本文提出一個首創、有效的演算法來解決這個問題。首先，我們利用K-mean方法將三維模型的頂點分成若干組。接著，我們藉由平移處理，修正頂點座標，藉此減少還原所可能產生的誤差。最後，我們將機密三維多邊形模型分享為 n 份影子資訊，只要取得其中任 r 份影子資訊 ($r \le n$) ，即可還原回具有較低失真率，成圖時具有高圖像品質的三維模型。實驗結果證實我們的演算法可以在進行漸進式還原三維模型時，有效的減少座標值的誤差，獲得失真率較少的三維模型與具備較佳圖像品質的成圖影像。我們的演算法仍能維持機密分享的優點，能兼顧達成機密三維模型分享的品質、可靠性及安全性。

關鍵詞：三維模型、漸進式、機密分享、資訊隱藏

Abstract

3D models have been widely used in many applications such as modern movies and computer games. This raises an important issue for secretly sharing 3D models and progressively retrieving them effectively. Current 3D model secret sharing algorithms is not in a progressive manner, and they suffer a common drawback not able to progressively retrieve a 3D model with good quality. In this paper, we propose an effective algorithm to overcome this drawback. Our algorithm first classifies the vertices of the input 3D model into several groups using the K-mean technique. For vertices in each group, we translate these vertices respect to the centroid of the corresponding groups. Finally, we encode the translated vertices into *n* shadow data, allowing to retrieve the original 3D model with good quality given any *r* (r<=n) shadow data. Experimental results demonstrate the feasibility. Our algorithm is capable of retrieving low distortion 3D models as well as producing rendered images with good quality.

Keywords: three-dimensional models, progressive, secret sharing, steganography. three-dimensional models, progressive, secret sharing, steganography.

1. 簡介

隨著科技的進步，繁瑣的實體郵寄流程逐漸被網際網路所取代。越來越多人選擇在網路儲存與分享機密資訊以求降低成本、減少時間。如何保護這些機密資訊就顯得十分重要，也是一個重要的研究主題[[23]]。

機密資訊分享(secret sharing)技術提供一個兼顧分散儲存及保密的方法。藉由將機密資訊(secret data)分割為 *n* 份影子資訊(shadow data)，唯有取得其中任意 *r* 份以上的影子資訊(r≤n)才可還原得到原本的機密資訊；若只取得 r 份以下的影子資訊，則無法獲得任何機密資訊。大部分的機密分享演算法都屬於二維，也

就是針對影像(images)媒體做機密分享[[5],[11],[14],[24],[27]-[28]],僅有少部分的演算法針對三維模型(three-dimensional models)做機密分享[[26]]。

機密資訊分享技術更進一步的發展是將其變成具有漸進式(progressive)的能力。藉由漸進式演算法的概念，先對機密資料重新編碼，再分割成 n 份影子資訊。如此一來，當取得其中任 k 份以上的影子資訊($k \le r$)即可還原部分的資訊。這種還原動作可以持續進行，直到 k 達到其上限 r 為止。以漸進式的方式還原機密資訊適合快速的得知「大輪廓」的機密資訊，亦即此法可以在影子資訊少於 r 份時，就開始還原並持續進行還原的動作，直至還原出全部的機密資訊。

隨著三維繪圖技術的蓬勃發展，許多領域開始廣泛運用三維模型；例如，電腦遊戲全面使用三維立體繪圖，電影界也利用三維繪圖創造出各種超現實的場景與生物，現在甚至連主角人物都可能是由三維模型所繪製的。有鑑於三維模型之運用趨勢及價值，且三維模型都需要做成像(rendering)的動作，故研究具備漸進式三維模型分享演算法將是一個重要的研究課題[[1],[7],[10]]。然而，就我們所知，文獻上並無這類的演算法。

一般的三維模型主要可以分為兩類——三維點模型以及多邊形三維模型。三維點模型僅具有頂點(vertices)座標的資訊；而多邊形三維模型則除了頂點的資訊以外，還包含了面(faces)索引的資訊。無論是哪種類型，由於座標資訊都是浮點數的特性，直接套用傳統針對影像的二維漸進式機密分享演算法[[4],[12]-[13],[15]]於三維模型，會因為還原座標值的誤差過大，導致漸進式還原的模型變形嚴重[[21],[29]]。經過成圖後，其圖像品質低劣，無法由影像中視覺出原始三維模型的幾何特性。為解決上述問題，本論文提出一個首創的漸進式機密三維模型分享演算法。這個演算法不僅能為三維模型提供一個新的漸近式機密分享機制，也能夠在每次漸進式的還原階段中，獲得較少誤差的三維模型；經過成圖後這些三維模型能展現良好的圖像品質，能在視覺化時，忠實顯示三維模型的幾何特質[[8],[16],[18],[20]]。

我們的演算法依據空間關係先將三維模型的頂點分為多個群集；我們將頂點座標值轉換為正整數，並以座標調整的技巧，縮減頂點座標值之數值長度；為避免浮點數運算所造成誤差，我們是以大數運算的方式進行座標值轉換。我們對頂點座標重新編碼後就可以取得影子資訊。這種作法讓我們得以在進行漸進式還原三維模型時，有效的減少座標值的誤差，獲得誤差較少的三維模型與具備較佳圖像品質的成圖影像。我們的演算法藉此漸進式的方式仍能維持機密分享的優點，能兼顧達成機密三維模型分享的品質、可靠性及安全性[[3][6],[17],[25]]。

本文架構如下：第二節回顧相關文獻；第三節敘述我們提出的演算法；第四節說明實驗結果。最後，第五節提出結論與未來工作。

2. 相關文獻

本節將回顧機密分享演算法。由於文獻上並無漸進式機密三維模型分享演算法，因此

我們將轉而討論二維影像漸進式機密分享演算法。

機密分享的概念亦即所謂*(r; n)*門檻法，最早是由 Blakley[[2]]及 Shamir[[22]]兩位學者在傳統的二維影像上分別提出。*(r; n)*門檻法透過多項式的分解，將機密資訊分享成 *n* 份看似無意義的影子資訊。當取得其中的任 *r* 張以上的影子資訊(r≤n)，即可還原回原始機密資訊。然而，他們提出的方法，一個方程式只能處理一個單位的資料，是故產生的影子資訊將與機密資訊一樣大，結果造成儲存空間的浪費。

Thien 及 Lin 改良了 Blakley 及 Shamir 所提出的*(r; n)*門檻法，縮小了影子資訊的儲存大小[[24]]。最主要的改良是將原本一個方程式僅處理一個單位的資料之模式，發展為一個方程式處理 *r* 個單位的資料，如此，將影子資訊的大小縮小為機密資訊的 *1/r*，大幅地降低影子資訊的儲存空間。

二維影像漸進式機密分享演算法的概念是由 Chen 和 Lin[[5]]提出。他們以*(r; n)*門檻法為基礎，再加入了 *k* 門檻值，*k* 即為訂定各階還原之門檻值 $r_1,r_2,...,r_k$，其關係為 $r_1 < r_2 < ... < r_k = r$，其中 $r_k = r$ 就是完全還原。隨後，將資訊分成數個區塊，每個區塊內的資訊皆由最高位元至最低位元重新排列，求得特徵值。將特徵值分別以門檻值 r_1 至 r_k 分別進行機密分享。當僅取得 r_1 影子資訊時，即可還原資訊中最重要的幾個位元，隨著獲得的影子資訊，可還原出的位元數越多，越接近原始的資訊。

3. 漸進式機密三維模型分享演算法

本節將詳細介紹「漸進式機密三維模型分享」演算法，圖 1 即為整體之流程。全部共分為四個步驟：1. K-mean 方法分群頂點座標值、2. 模型各群座標修正、3. 頂點座標值轉換及分組、4. 漸近式三維模型分享。3.1 節說明 K-mean 分群如何使用在此演算法中，以及接著利用大數運算修正圖形的座標；3.2 節將詳細說明機密多邊形模型資訊在透過漸進式機密分享後產生 *n* 筆影子資訊之過程；3.3 節說明在取得 $r_1,r_2,...,r_k$ 個偽裝模型後，依符合條件來還原部份至全部的原始機密三維模型。

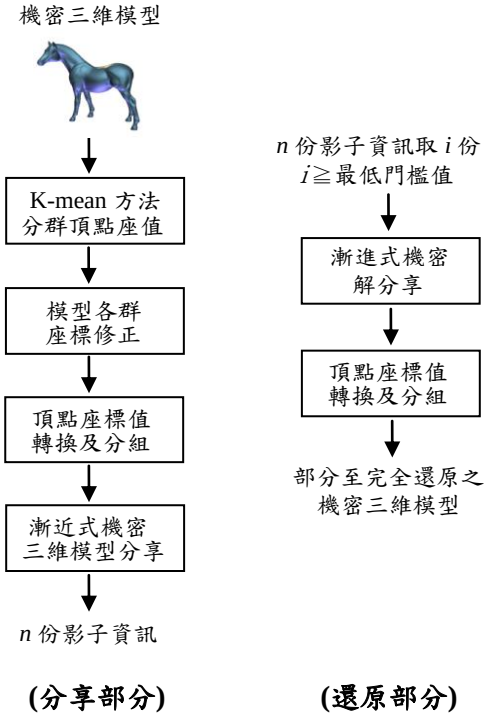


圖 1 機密三維多邊形模型之分享與流程圖

3.1 K-mean 多邊形模型頂點分群

一個三維多邊形模型所包含的頂點座標往往分佈於分散的空間中，因而各個頂點座標

數值的是分佈在一個極大的範圍，若直接利用傳統的漸進式機密分享演算法來處理三維模型的資料，在日後以漸進方式還原時，還原得到的頂點座標值與原始座標值將產生很大的誤差，而使得模型形狀嚴重地變形，使用者將很容易察覺模型品質的降低。為了縮小這個誤差，我們必須將所有的頂點座標值加以縮減，如此一來漸進式還原時的座標值誤差就會減少，使得模型的變形程度也跟著降低[[9],[19]]。

我們利用 K-mean 演算法，根據頂點座標間的空間關係將其分為數個群集，並配合座標調整，使各個頂點座標值調整為較小的數值。如此一來日後漸進式還原時的誤差程度也會縮小，而獲得較佳的模型還原品質。

首先，由三維多邊形模型頂點中隨機挑出 n 個點作為基準點，並算出每個頂點座標與各個基準點之間的距離 D_i。再依據每個點座標與其距離最近的基準點而分成 k 個群集。接著對每個群集求出重心 G_i，並以此重心 G_i 作為新的基準點，重複以上步驟直到群集成員不再改變為止。如圖 2 所示

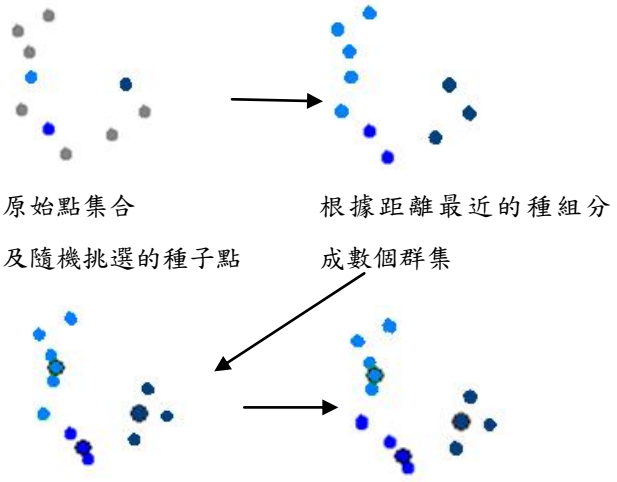


圖 2 K-mean 分群演算法示意圖

接著，我們對每個群集求出重心 G，然後平移整個群集的座標使 G 成為該群集的新原點。這個方式可以將每個頂點座標之數值長度減為最小，而不影響群組內各點之相對位置關係。即對每一群組之重心 G(G_x, G_y, G_z) 及群組中每個點座標 P (P_x, P_y, P_z)，根據方程式 1 將 P 平移至相對於重心之適當位置 P'。

$$P' = (P_x - G_x, P_y - G_y, P_z - G_z) \qquad (1)$$

為避免浮點數運算產生之誤差，我們將浮點數分解為數字部份及指數部份以避免對浮點數做運算，即可以整數運算取代浮點數運算，而防止產生誤差。

3.2 影子資訊與分享技術

在進行漸進式機密分享演算法時，必須對各座標值按比例切割為數個部分，進而達成漸進式的分享。但三維多邊形模型的頂點座標值往往是具有多個小數位數之有號數字，並不適合做比例切割，所以我們以頂點為單位，找出各頂點座標 x、y 及 z 之值轉換為有號整數所需移動的小數點位數最大值 m_i，並以 m_i 為基準，將頂點中各座標以字串處理的方式去除小數點，並記錄其原始位置，使頂點座標值皆轉換為有號整數，如圖 3 所示。

x	y	z	須移動之位數
-2372.27	1449.71	153.132	3
-1037.78	3047.39	-682.49	2
-979.144	390.65	-6.96055	5
…	…	…	…

x	y	z
-2372270	1449710	153132
-103778	304739	-68249
-97914400	39065000	-696055
…	…	…

圖 3 頂點座標值轉換示意圖。

為了將頂點座標值分割後代入分享方程式，我們把所有頂點座標 x、y 及 z 之值以 2 進位表示後，再藉由方程式 2 轉為無號 2 進位整數之字串。用意在於將正號數值的座標值尾端補上一個 0 記；在負號數值的座標值尾端補上一個 1。並為了方便分享的進行，對頂點座標依其 m_i 值進行分組。C 為 2 進位表示之座標值。

$$C' = \begin{cases} 2|C| & if \quad C \geq 0 \\ 2|C|+1 & if \quad C < 0 \end{cases} \tag{2}$$

接下來，對分組後的頂點座標值進行漸進式分享。首先訂出需產生的影子資訊數 n、完全還原的門檻值 r，以及漸進式的還原階段數 k；再依 k 值選出各階還原之門檻值 $r_1, r_2, ..., r_k$，其關係為 $r_1 < r_2 < ... < r_k = r \leqq n$。之後將頂點座標值分成 k 個還原階段，各個階段之資訊還原量由使用者自定；面索引值則在第一個還原階段即須完全還原；在分享之後得到 n 份影子資訊，而拿到 r 份則可以完全還原。因為各頂點座標值的 k 個部分與面索引值在進行漸近式分享演算法時皆須模一個質數，而為了有效地減小在分享之後產生的影子資訊量，我們把 x、y、z 三軸之座標值分開處理。然後，把所有 2 進位表示之頂點座標值，依設定之還原比例分割為 k 個長度不定的 2 進位字串，所以模型中任一頂點 P_i 將具有 $3k$ 個資料部分。

將各軸頂點座標值的 k 個還原階段為單位，分別找出 x、y 及 z 各軸第 k 階段中大於其最大頂點座標值之最小質數 P_k^x、P_k^y、P_k^z。並以此 $3k$ 個質數決定各個部分在分享後之影子資訊所需使用的最大儲存位元長度，因此可減少分享後之影子資訊量。接著，我們對頂點座標 x、y、z 的 k 個還原階段各定義一個介於 r-1 到 1 階的多項式，如方程式 3。

$$\begin{aligned} q_j(x) &= (a_0 + a_1 x + ... + a_{r-1} x^{r-1}) \bmod P_k \\ q_{j-1}(x) &= (a_0 + a_1 x + ... + a_{r-2} x^{r-2}) \bmod P_{k-1} \\ &\cdots \\ &\cdots \\ q_1(x) &= (a_0 + a_1 x) \bmod P_1 \end{aligned} \tag{3}$$

其中多項式的階層數是依照使用者定義之各階還原所需之影子資訊數量而定。而面索引值的部份，因其必須在第一階段就完全還原，所以是與點座標值的第一個還原階段使用相同階層的方程式進行分享。$a_0, a_1, \ldots,$ 是為各軸第 k 個部分之數值，x 為產生影子資訊所需代入之鑰匙值，藉以計算出各個階段的 n 個影子資訊數值：$q_j(1), q_j(2), \ldots, q_j(n) \sim q_1(1), q_1(2), \ldots, q_1(n)$，如方程式 4。

$$\begin{aligned} q_j(x) &= (a_0 + a_1 x) \bmod P_k \\ q_{j-1}(x) &= (a_2 + a_3 x) \bmod P_{k-1} \\ &\cdots \\ &\cdots \\ q_1(x) &= (a_{r-2} + a_{r-1} x) \bmod P_1 \end{aligned} \tag{4}$$

最後，將所有使用同個鑰匙值產生的各個影子數值共同輸出後，即得到還原所需的 n 個影子資訊。

3.3 三維模型還原方法

此節描述將影子資訊將初始之三維模型漸進式還原的方法：一開始，假設拿到 i 個影子資訊，然後檢查之前定義的分享階段設定，

i 值是否滿足 k 個門檻值中之一；若滿足，則依其滿足的門檻值還原量把影子資訊中可還原的點座標值部分代回相對應的方程式；如果沒有滿足任一個門檻值，就無法進行任何的還原動作。而面索引值的部分，因分享設定的關係，只要影子資訊數量達到最低門檻值，就將它完全還原。

各個影子資訊其鑰匙值及達到還原門檻值之各部分影子數值 $q_j(1), q_j(2), \ldots, q_j(n)$ ~ $q_1(1), q_1(2), \ldots, q_1(n)$與各方程式所須之質數 P_1 ~ P_k 代回方程式後，則得到第 k 階段還原所得之部分原圖點座標值與完全還原之面索引值資訊。

最後，將所得之圖形各群重心移回原始位置，點座標值位元長度補回與原始長度相同，依最後一個 bit 值還原其原始之正負號及移回小數點至原本位置，與面索引值一起輸出後，就拿到第 k 階段還原部分資訊之三維模型。

4. 實驗結果

本節將說明並分析的實驗成果。我們使用 Ruby 語言作為開發語言，測試平台為 Intel Core 2 Duo 2.33 GHz 的 CPU 及 2 GB 的記憶體，作業系統為 Windows XP Professional。以下將使用模型圖例說明此演算法實際進行漸進式機密分享的結果。

我們進行實驗的設定是將機密三維模型分為三個階段進行分享，第一階段還原 33% 點資訊、第二階段 66%、第三階段完成無失真還原模型。我們以 2、3、4 為此三階段之門檻值。

圖 4、圖 5 為二組不同之漸進式機密三維模型分享範例，圖 4(a)、圖 5(a)分別為機密三維模型：尼克及恐龍模型，尼克之頂點數有 7049 個，三角面數有 13869 個；恐龍之頂點數有 56194 個，三角面數有 112384 個。圖 4 為使用我們演算法（分 30 群、做座標修正處理）將模型還原 33%與 66%的結果。我們也比較不使用我們演算法(不分群、不做座標修正處理)的結果。由圖可知，我們的演算法在漸進式還原過程中，能在僅還原 33%原始三維模型時，即顯示優異的成圖品質。反之，不使用我們的演算法要在還原至少到 66%時，才能視覺出原始三維模型的幾何特性。

圖 5 是恐龍模型的實驗結果。同樣的，我們列出使用我們的演算法與不使用我們的演算法的結果。由圖可知，使用我們的演算法仍具有極大的優勢：我們的演算法仍舊能在僅還原 33%原始三維模型時即顯示優異的圖像品質。圖 4 與圖 5 之結果除了顯示顯示，在進行了分群座標修正後再漸近式分享，將可大幅改善在較低還原階段的結果圖形，也就是在同樣較低的資訊量之下有較好的還原效果。

表 1 顯示還原的失真率。這是使用我們的演算法(分 30 群、做座標修正處理)來還原三維模型後與原始三維模型的比較。我們分別比較 33%還原與 66%還原三維模型之失真率。由於我們的演算法是無失真的漸進式機密分享，故當全部還原時，沒有任何失真，故其失真率為 0.0。我們以 root mean square error (RMSE)量測失失真數值，其幾何意義代表某個三維模型中，還原後的模型與原始模型比較下，平均一個頂點的差異距離(distortion distance)。由於每個三維模型大小不一，必

須有統一的標準才能比較 RMSE 數值代表之優劣。因此，我們先求出每個三維模型的界線容積(Bounding Volume)，此代表一個包含三維模型並有最小體積的長方體。然後，我們求出其斜對角線長度(Bounding Volume Diagonal, BVD)。將上述 RMSE 對比於 BVD，則可以在相同基準下，瞭解每個三維模型內，平均每個頂點差異距離與斜對角線長度的比例，也就是失真率。較小的數據代表具有較小的失真率。

表之數據顯示：如預期般，僅有 33%模型還原與 66%模型還原時，前者具有較高的失真率。然而，33%模型還原的失真率都在 $7.73x10^{-2}$ 以下，66%模型還原的失真率更小，都在 $1.45x10^{-2}$ 以下。尼克模型甚至僅有 $0.30x10^{-4}$。這些數據顯示每個測試模型在各個還原階段，都具有很小的失真率。此也代表我們的演算法在還原時，確實能有效的獲得具有較小誤差的三維模型。這些模型經過成圖後，也顯示出具有優異的圖像品質，如圖 4、5 所示。 最後，此表也列出模型分享時間，其數值介於 24～242 秒間。由表之數據可以看出，如預期般，分享時間與三維模型之幾何複雜度成正比。

圖 6、圖 7 顯示不同的分群，在漸進式機密分享時，對模型成圖品質的影響。圖 6(a)為花結模型，其頂點數有 23232 個，三角面數有 46464 個。圖 7(a)為馬之模型，其頂點數有 48485 個，三角面數有 96966 個。圖 6(b)、圖 6(c)、圖 6(d)及圖 7(b)、圖 7(c)、圖 7(d)是將機密模型分為 5、15、25 群並做座標修正處理後做機密分享，並還原 33%原始模型之結果。圖 6、7 顯示不同模型複雜度需要考慮不同的分群策略。花傑模型僅需分為 5 群，在還原時即具有良好的圖像品質，但馬模型即使分為 25 群後，在成圖時，依稀仍可察覺出頸部表面的不平滑。

從圖 8 分析不同分群數量下，對模型點做座標修正之影響。座標修正代表將頂點相對平移該群之重心位置。由數據可知，進行分群越多並對座標修正後，三維模型中之所有頂點座標相對於重心的平均距離也逐漸減小。在此情況下，每個頂點座標之數值變得較小。在做模型分享時，我們切割同樣比例位元數之頂點座標值做漸進式分享，故並無任何影響。但是，在模型還原的各階段，分群並做座標修正則會使得我們能得到更精確的點頂座標，此代表這些還原的頂點座標 V'之數值更接近於原來模型的頂點座標 V。故在成圖時，在首次的還原階段即可得到很好的圖像品質，如圖 4(e)與 5(e)。相對的，如果分群數較少甚或沒有進行分群時，還原的數值則較為不精確，導致還原頂點的座標值 V'與原始點頂的座標值 V 相對仍存有較大的差異。此當然增加模型的失真率，相對的也降低模型成圖的影像品質，如圖 4(b)與 5(b)所示。

圖 9 顯示分群並做座標修正後，還原 33%模型時，成圖品質之影響 PSNR (Peak Signal to Noise Ratio)數值。兩個影像之 PSNR 值越高代表此兩個二個影像之間的相似度越高。由圖可知，較高的分群數可提高 PSNR 數值，增加還原模型成圖與原始模型成圖的相似度，提高成圖的品質。此外，大部分的模型之 PSNR 數值隨著分群數之增加呈現穩定成長。然而，尼克模型之 PSNR 數值在初期即有大幅提高之

現象。此乃因為此模型的頂點座標本來就與重心點有相當大的差異，所以只要做過座標修正後，即使不分群頂點與重心之平均距離就已大幅減少，故 PSNR 數值在不分群不做座標修正(分群數為 0)與僅作座標修正（分群數為 1）間有大幅的成長。

最後，圖 10 顯示分群並做座標修正後，還原 33%模型後，各模型在不同分群數量下的失真率。圖中數據顯示，經過分群並做座標修正處理後，還原模型的失真率隨著分群數量的提高而逐漸降低。此外，高分群數量具有更小的失真率。由此數據證實，藉由提高分群數並做座標修正，可降低還原三維模型之失真率。然而，此舉所必須付出之代價為需要耗費更多的時間作分群的處理。此也代表吾人必須在提高分群數量與耗費之時間尋求一個平衡點。

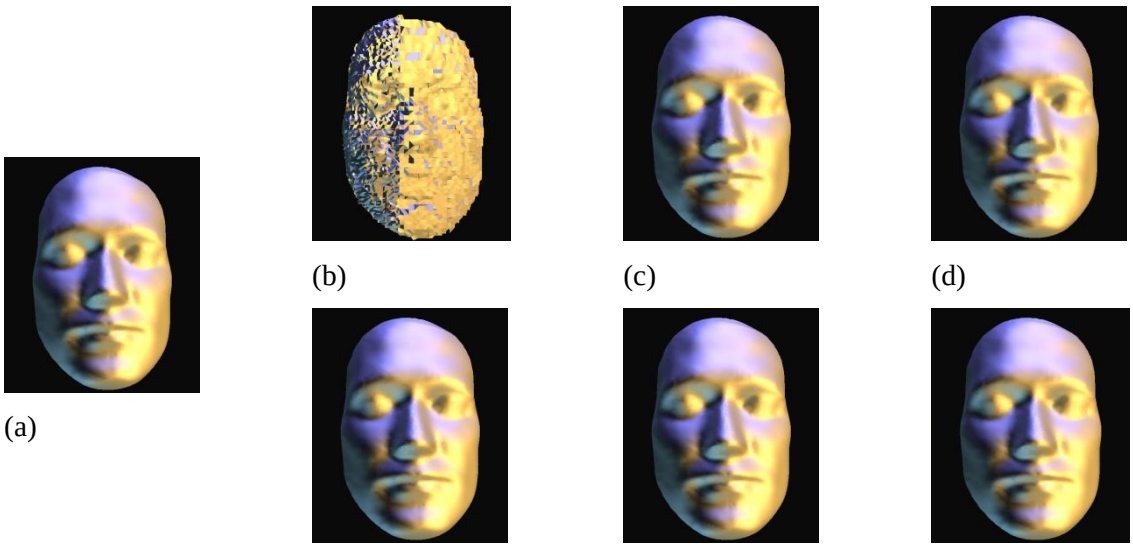
5. 結論與未來工作

總結本研究，我們提出一個文獻上首創的漸進式機密三維模型分享演算法。這個演算法在進行漸進式還原三維模型時，有效的減少座標值的誤差，獲得誤差較少的三維模型。此演算法也因此能產生較佳圖像品質的成圖影像。我們的演算法雖藉漸進式的方式，但仍能維持機密分享的優點，能兼顧達成機密三維模型分享的高品質、可靠性及安全性要求。

未來工作方面，我們認為可朝以下二種方向前進：第一、縮短分享的時間。由於模型的幾何複雜度會影響分群的時間，未來可進行不同複雜度下三維模型最佳化分群的研究，藉此減少在分群時所花的時間，以利進行分享。第二、縮減影子資訊量：目前本演算法並未對影子資訊做任何壓縮處理；未來可適當的將影子資訊進行壓縮，以降低影子資訊之大小，此將更有利於增加演算法在網際網路上做機密三維模型分享的傳輸效率

致謝

本研究承蒙國科會之經費補助(95-2815-C-005-028-E、95-2815-C-005-029-E、95-2815-C-005-027-E)，謹此致謝。



(a) (b) (c) (d) (e) (f) (g)

圖 4 尼克的實驗結果：(a)為原始機密三維模型；(b)、(c)、(d)為不進行分群與座標修正處理的

成圖結果。在成圖時，模型分別還原為原始三維模型點資訊的 33%、66%、100%。 (e)、(f)、(g)為使用我們演算法，分 30 群並做座標修正處理的結果。模型也是分別還原為原始三維模型點資訊的 33%、66%、100%。其中 (d)、(g)在成圖時，其三維模型與原始三維模型(a)完全相同，代表無失真的完全回復三維模型。

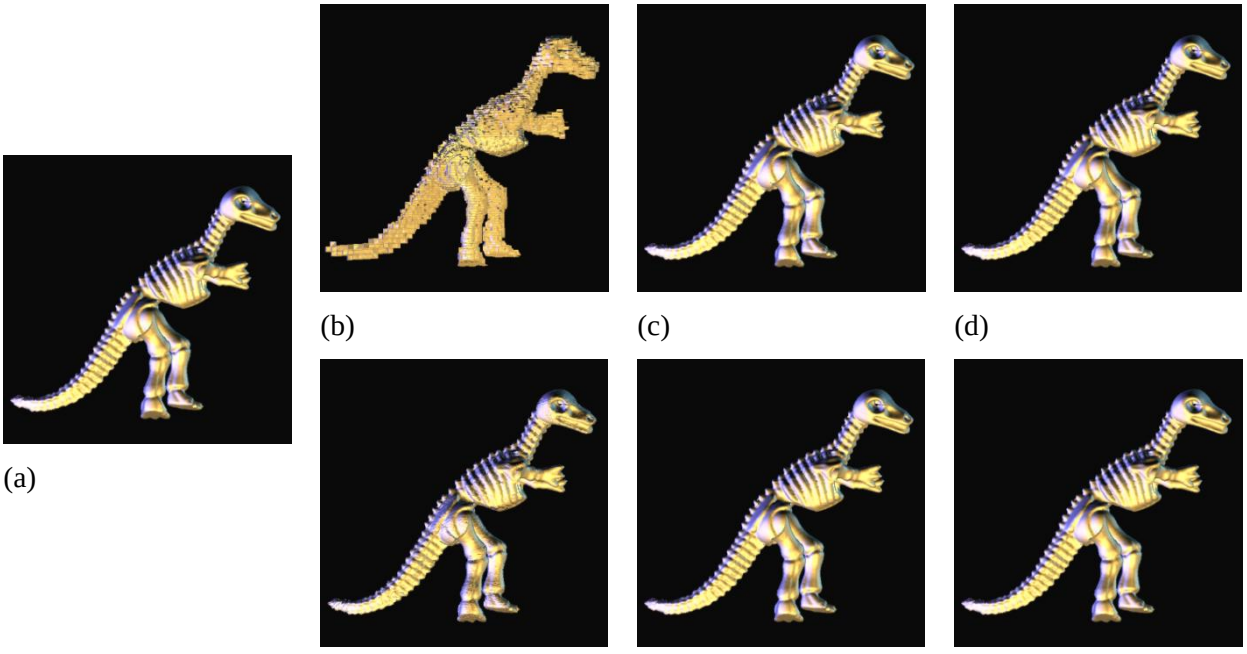
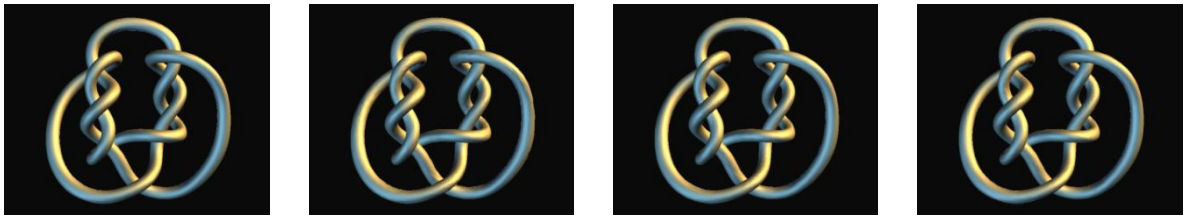


圖 5 恐龍的實驗結果：(a)為機密三維模型；(b)、(c)、(d)為不進行分群座標修正處理分享，分別還原點資訊 33%、66%、100%之結果圖形， (e)、(f)、(g)為分 30 群座標修正處理分享，分別還原點資訊 33%、66%、100%之結果圖。其中(a)、(d)、(g)完全相同。

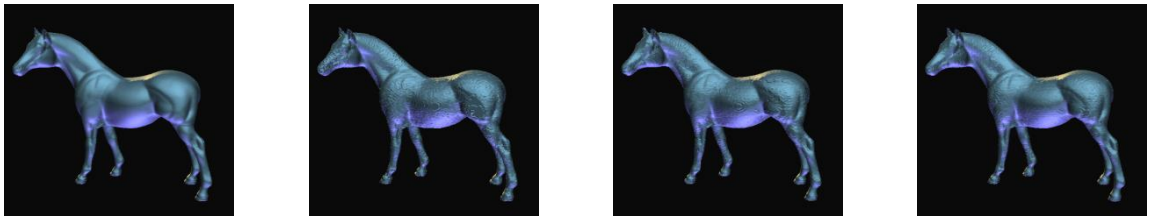
表一、 使用我們的演算法所還原之三維模型與原始三維模型之失真率比較

三維多邊形模型		點數	面數	還原 33% 失真率 (RMSE/BBD)	還原 66% 失真率 (RMSE/BBD)	分享時間 (second)
兔子		34834	69451	6.14×10^{-2}	0.98×10^{-2}	150.69
恐龍		56194	112384	7.73×10^{-2}	1.45×10^{-2}	242.11
馬		48485	96966	6.25×10^{-2}	1.08×10^{-2}	196.30
花結		23232	46464	1.45×10^{-2}	0.2×10^{-2}	86.55
尼克		7049	13869	5.89×10^{-4}	0.30×10^{-4}	24.42



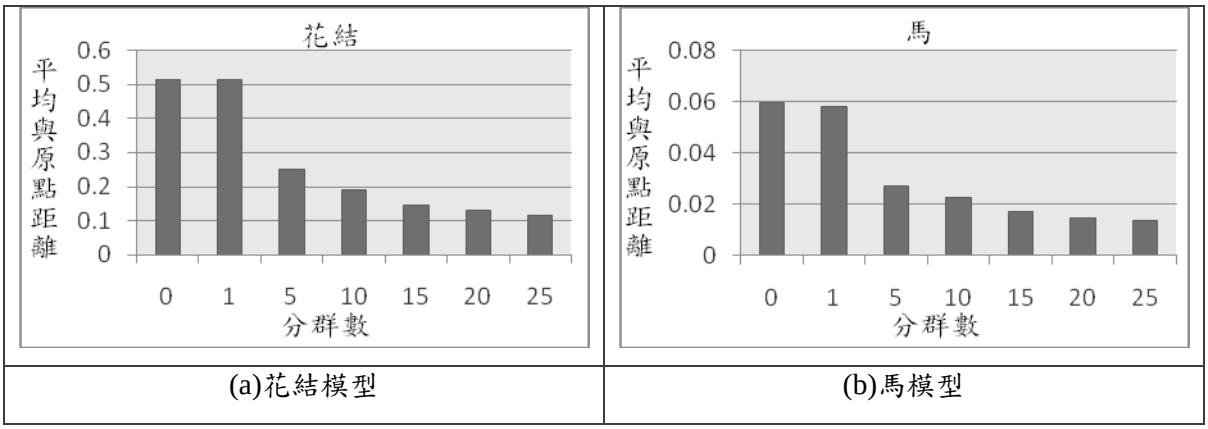
(a) (b) (c) (d)

圖 6 不同的分群，在漸進式機密分享還原 33%時，對模型成圖品質的影響。花結模型的實驗結果：(a)為原始機密三維模型; (b)、(c)、(d)分別為進行分 5 群、分 15 群、分 25 群並做座標修正分享後還原之結果。



(a) (b) (c) (d)

圖 7 不同的分群，在漸進式機密分享還原 33%時，對模型成圖品質的影響。馬模型的實驗結果：(a)為原始機密三維模型; (b)、(c)、(d)分別為進行分 5 群、分 15 群、分 25 群並做座標修正分享後還原之結果。



(a)花結模型 (b)馬模型

圖 8 分群數與座標修正後頂點座標與各群重心之平均距離分佈圖。

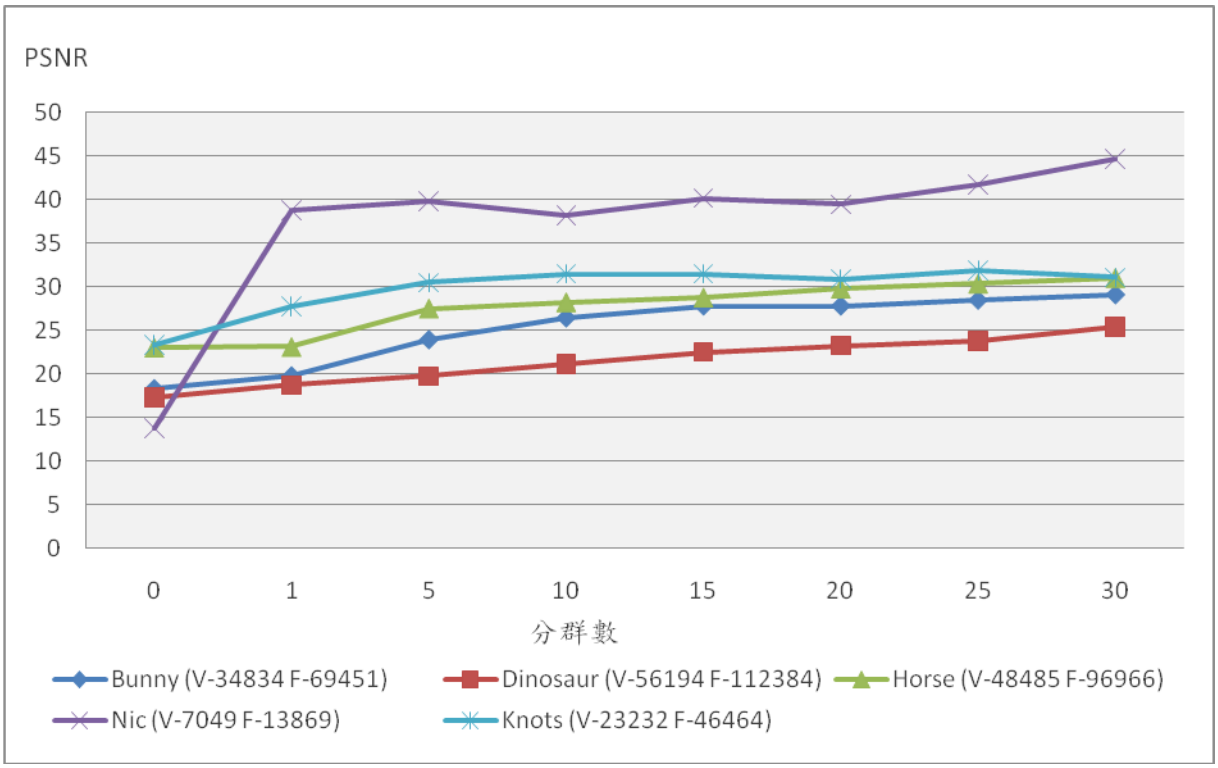


圖 9 分群並做座標修正後，還原 33%模型的 PSNR 數據圖。分群數 0 表示不進行分群、不做座標修正；分群 1 代表僅作座標修正。

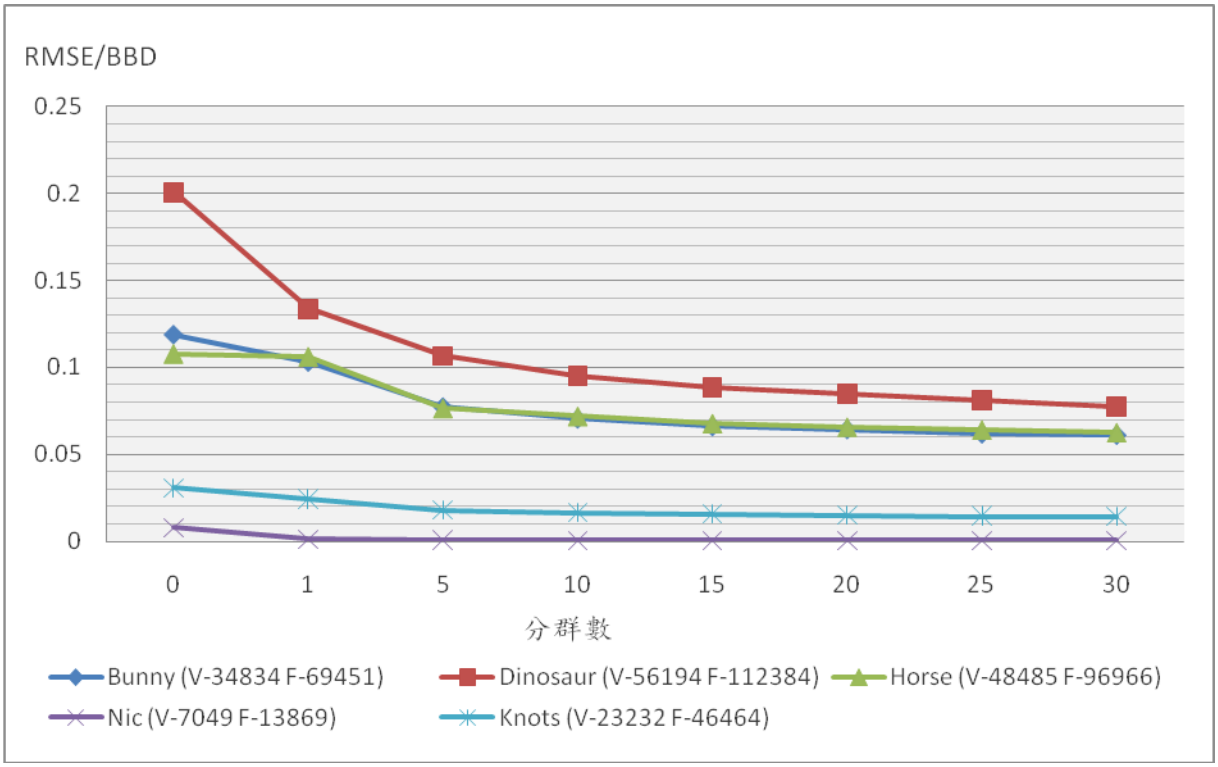


圖 10 分群並做座標修正後，還原 33%模型的失真率數據圖。分群數 0 表示不進行分群、不做座標修正；分群 1 代表僅作座標修正。

表二 、不同分群數並做座標修正之分享時間比較。此處無處理代表為不分群、不做座標修正。

三維多邊型模型		分享時間(second)							
		無處理	分 1 群	分 5 群	分 10 群	分 15 群	分 20 群	分 25 群	分 30 群
兔子		48.516	49.157	64.453	81.531	99.516	115.171	133.063	150.687
恐龍		89.391	91.391	126.891	136.438	162.775	189.65	217.04	245.602
馬		72.44	73.219	82.781	104.156	125.906	149.985	173.718	196.297
花結		30.703	30.875	35.797	46.563	56.343	66.063	74.984	86.547
尼克		7.329	8.281	10.453	12.875	15.938	18.781	21.656	24.422

參考文獻

[1] Arkin, E. M., Held, M., Mitchell, J. S. B., and Skiena, S., "Hamiltonian Triangulations for Fast Rendering," ***Visual Computer,*** Vol. 12, No. 9, pp. 429-444, 1996.

[2] Blakley, G. R., "Safeguarding Cryptographic Keys," ***In Proceedings of AFIPS' 1979 National Computer Conference***, Vol. 48, pp. 313-317, 1979.

[3] Cayre, F., and Macq, B., "Data Hiding on 3-D Triangle Meshes," ***IEEE Transactions on Signal Processing***, Vol. 51, No. 4, pp. 939–949, 2003.

[4] Chang, C. C. and Chuang, J. C., "An Image Intellectual Property Protection Scheme for Gray-Level Images Using Visual Secret Sharing Strategy," ***Pattern Recognition Letters***, Vol. 23, pp. 931-941, 2002.

[5] Chen, S. K. and Lin, J. C., "Fault-Tolerant and Progressive Transmission of Images," ***Pattern Recognition***, Vol. 38, pp. 2466-2471, 2005.

[6] Cheng, Y. M. and Wang, C. M., "A High-Capacity Steganographic Approach for 3D Polygonal Meshes," ***Visual Computer***, Vol. 22, pp. 845–855, 2006.

[7] Chow, M., "Optimized Geometry Compression for Real-Time Rendering," ***In Proceedings of the IEEE Visualization '97***, pp. 347–354, 1997.

[8] Cotting, D., Weyrich, T., Pauly, M., and Gross, M., "Robust Watermarking of Point-Sampled Geometry," ***Proceedings of International Conference on Shape Modeling and Applications***, pp. 233-242, 2004.

[9] Deering, M., "Geometry Compression," ***Proceedings of SIGGRAPH'95***, pp. 13–20, 1995.

[10] Evans, F., Skiena, S. S., and Varshney, A., "Optimizing Triangle Strips for Fast Rendering," ***Proceedings of IEEE Visualization***, pp. 319–326, 1996.

[11] Feng, J. B., Wu, H. C., Tsai, C. S., and Chu, Y. P., "A New Multi-Secret Images Sharing Scheme Using Largrange's Interpolation," ***Journal of Systems and Software***, Vol. 76, pp. 327-339, 2005.

[12] Hou, Y. C., "Visual Cryptography for Color Images," ***Pattern Recognition***, Vol. 36, pp. 1619-1629, 2003.

[13] Lin, C. C. and Tsai, W. H., "Visual Cryptography for Gray-Level Images by Dithering Techniques," ***Pattern Recognition Letters***, Vol. 24, pp. 349-358, 2003.

[14] Lin, C. C. and Tsai, W. H., "Secret Image Sharing with Steganography and Authentication," ***Journal of Systems and Software***, Vol. 73, pp. 405-414, 2004.

[15] Lukac, R. and Plataniotis, K. N., "Bit-Level Based Secret Sharing for Image Encryption," ***Pattern Recognition***, Vol. 38, pp. 767-772, 2005.

[16] Ohbuchi, R., Mazuda, H., and Aono, M., "Watermarking Three-Dimensional Polygonal Models," ***Proceedings of the ACM Multimedia '97***, pp. 261-272, 1997.

[17] Ohbuchi, R., Mazuda, H., and Aono, M., "Data Embedding Algorithms for Geometrical and Non-Geometrical Targets in Three- Dimensional Polygonal Models," ***Computer Communications,*** Vol. 21, pp. 1344-1354, 1998.

[18] Ohbuchi, R., Mazuda, H., and Aono, M., "Watermarking Three-Dimensional Polygonal Models Through Geometric and Topological Modifications," ***IEEE Journal on Selected Areas in Communications***, Vol. 16, No. 4, pp. 551-560, 1998.

[19] Peng, J., Kim, C. S., and Kuo, C. C. J., "Technologies for 3D Mesh Compression," ***Journal of Visual Communication and Image Representation***, Vol. 16, No. 6, pp. 688–733, 2005.

[20] Praun, E., Hoppe, H., and Finkelstein, A., "Robust Mesh Watermarking," ***Proceedings of SIGGRAPH'99***, pp. 49-56, 1999.

[21] Rencher, A. C., "Methods of Multivariate Analysis," ***Wiley***, 2nd ed., 2002.

[22] Shamir, A., "How to Share a Secret," ***Communications of the ACM***, Vol. 22, No. 11, pp. 612-613, 1979.

[23] Stallings, W., "Cryptography and Network Security," ***Methods of Multivariate Analysis***, 2nd ed., Wiley, Inc., 1999.

[24] Thien, C. C. and Lin, J. C., "Secret Image Sharing," ***Computers & Graphics***, Vol. 26, pp. 765-770, 2002.

[25] Wang, C. M. and Cheng, Y. M., "An Efficient Information Hiding Algorithm for Polygon Models," ***Computer Graphics Forum,*** Vol. 24, No. 3, pp. 591–600, 2005.

[26] Wang, C.M., Cheng, Y. M., Wu, K. C., Tao, C. W., Tseng, Y. T., and Liao, Y. K., "A Secret 3D Polygonal Model Sharing Algorithm with Steganography," Journal of Engineering, National Chung Hsing University, Vol. 18, No. 1, pp. 41-53, 2007.

[27] Wang, R. Z. and Su, C. H., "Secret Image Sharing with Smaller Shadow Images," ***Pattern Recognition Letters***, Vol. 27, pp. 551-555, 2006.

[28] Wu, Y. S., Thien, C. C., and Lin, J. C., "Sharing and Hiding Secret Images with Size Constraint," ***Pattern Recognition***, Vol. 37, pp. 1377-1385, 2004.

[29] Xiang, X., Held, M., and Mitchell, J., "Fast and Efficient Stripification of Polygonal Surface Models," ***ACM Symposium on Interactive 3D Graphics Proceedings***, pp. 71–78, 199