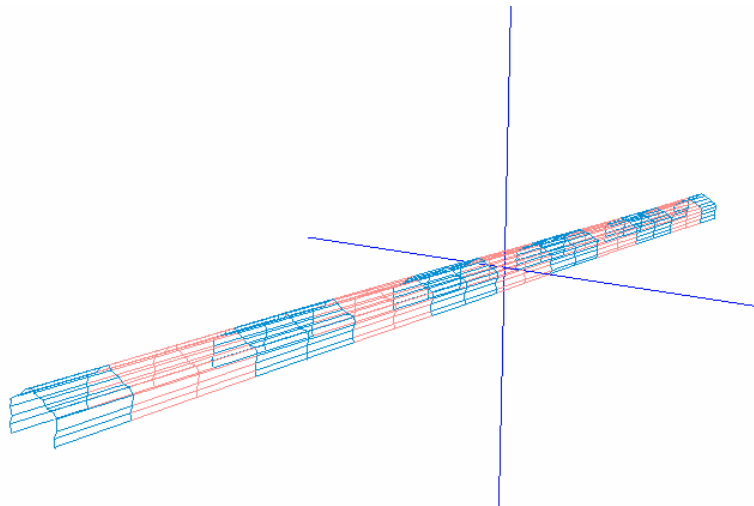


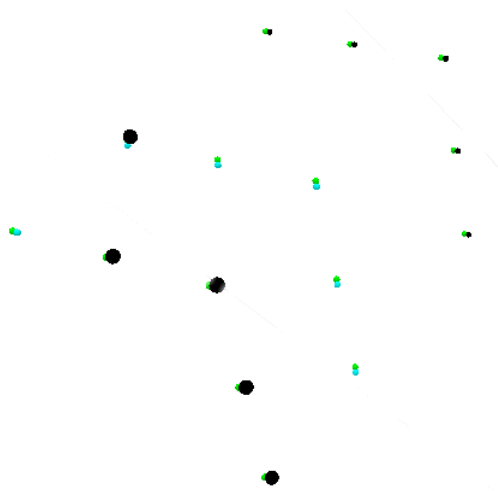
(12) 隧道三維變形與地質破壞模型之探討

目的

隧道施工之進度與安全管控，和複雜多變的地質狀況有著密不可分的關聯性。而傳統的監測資料如內空收斂及頂拱沉陷等觀測，只能量測隧道斷面兩測點間的相對變位量，對於靠近開挖面附近複雜的三向度問題，所能提供之預警訊息相當有限。根據國外學者的研究發現，開挖面附近的縱向變位，將受前方地質條件影響而有不同的反應，若能取得開挖面附近測點之絕對變位量，並將其彙整為易於解讀的影響趨勢線，則可據以預判開挖面前方的地質狀況，達到防災減災之目的。



以網格線的方式來隧道三維變形資料



用一系列的圓球來表示測點的相關資訊

● 部份程式碼

1. Visual Basic

```
Private Sub LoadDatabase()  
    Dim strIP As String  
    Dim strAccount As String  
    Dim strPassword As String  
    Dim strDatabase As String  
    Dim strTable As String  
    Dim strSQL As String  
    Dim strTemp1 As String  
    Dim index1 As Integer  
    Dim index2 As Integer  
    Dim index3 As Integer  
    Dim i As Integer  
    Dim sName As String  
    Dim x As Double  
    Dim y As Double  
    Dim z As Double  
    Dim size As Double  
    Dim color As Long  
    Dim data As String  
    Dim olddata As Collection  
  
    On Local Error GoTo errOut  
  
    Set olddata = m_data  
    Set m_data = New Collection  
  
    strIP = GetSetting("3D_Tunnel_Display_System", "DatabaseConfig", "txtIP")  
    strAccount = GetSetting("3D_Tunnel_Display_System", "DatabaseConfig", "txtAccount")  
    strPassword = GetSetting("3D_Tunnel_Display_System", "DatabaseConfig", "txtPassword")  
    strDatabase = GetSetting("3D_Tunnel_Display_System", "DatabaseConfig", "txtDatabase")  
    strTable = GetSetting("3D_Tunnel_Display_System", "DatabaseConfig", "txtTable")  
  
    Set myCon = New ADODB.Connection  
    myCon.ConnectionString = "driver={SQL Server};" & _  
        "server=" & strIP & ";" & _  
        "uid=" & strAccount & ";" & _  
        "pwd=" & strPassword & ";" & _  
        "database=" & strDatabase  
  
    myCon.ConnectionTimeout = 30  
    myCon.Open  
    strSQL = "select * from " & strTable  
    Set myRS = myCon.Execute(strSQL)  
  
    While Not myRS.EOF  
        x = val(myRS("x").value)  
        y = val(myRS("y").value)  
        z = val(myRS("z").value)  
        index1 = val(myRS("TimeID").value)  
        index2 = val(myRS("MileID").value)  
        index3 = val(myRS("MarkID").value)  
  
        myOriDataX(index1, index2, index3) = x  
        myOriDataY(index1, index2, index3) = y  
        myOriDataZ(index1, index2, index3) = z  
  
        intNoTime = index1  
        intNoMile = index2  
  
        'set defaults  
        i = i + 1  
        size = 0.2  
        color = D3DCOLORVALUEtoLONG(ColorValue4(0.4, CSng(BallR(0) / 255#), CSng(BallG(0) / 255#),  
CSng(BallB(0) / 255#)))  
        data = ""  
        sName = "1"  
  
        If index1 = 1 Then  
            AddEntry sName, x, y, z, size, color, data  
        Else
```

計算調整後之差異大小

```
tempX = (myOriDataX(index1, index2, index3) - myOriDataX(1, index2, index3)) * EScaleX  
tempY = (myOriDataY(index1, index2, index3) - myOriDataY(1, index2, index3)) * EScaleY  
tempZ = (myOriDataZ(index1, index2, index3) - myOriDataZ(1, index2, index3)) * EScaleZ
```

依調整差異大小決定位置

```
tempX = tempZ + myOriDataX(1, index2, index3)  
tempY = tempY + myOriDataY(1, index2, index3)  
tempZ = tempZ + myOriDataZ(1, index2, index3)
```

計算圓的位置

```
myPX(index1, index2, index3) = (tempX + myOriDataX(1, index2, index3)) / 2  
myPY(index1, index2, index3) = (tempY + myOriDataY(1, index2, index3)) / 2  
myPZ(index1, index2, index3) = (tempZ + myOriDataZ(1, index2, index3)) / 2  
size = (myPX(index1, index2, index3) - tempX) ^ 2 + (myPY(index1, index2, index3) - tempY) ^ 2 +  
(myPZ(index1, index2, index3) - tempZ) ^ 2  
size = Sqr(size)  
color = D3DCOLORVALUEtoLONG(ColorValue4(0.4, CSng(BallR(9) / 255#), CSng(BallG(9) / 255#),  
CSng(BallB(9) / 255#)))  
color = D3DCOLORVALUEtoLONG(ColorValue4(0.4, 1, 0, 0))  
  
x = myPX(index1, index2, index3)  
y = myPY(index1, index2, index3)  
z = myPZ(index1, index2, index3)
```

```
' AddEntry sName, x, y, z, size * 0.1, color, data  
End If  
myRS.MoveNext  
Wend
```

```
Set m_graphroot = Nothing  
Set m_quad1 = Nothing  
Set m_quad2 = Nothing  
Set m_XZPlaneFrame = Nothing
```

```
ComputeDataExtents  
BuildGraph  
RestoreDeviceObjects
```

```
Set olddata = Nothing  
Set myRS = Nothing
```

```
Exit Sub  
errOut:  
Set m_data = olddata  
MsgBox Err.Description, vbCritical, "連結資料庫發生錯誤"  
Set myRS = Nothing  
End Sub
```

- **時間**：民國 91 年 10 月到 92 年 10 月

- **參與人員**

1. 台灣科技大學營建系陳堯中教授
2. 台灣科技大學營建系呂守陞教授
3. 台灣科技大學營建系陳志南教授
4. 台灣科技大學工管系周碩彥教授
5. 台北科技大學陳立憲教授
6. 中國技術學院企管系林詩偉助理教授

- **本人工作內容**

1. 隧道三維變形與地質破壞資料之表達

- **開發工具及環境**

1. 作業系統：Window 98/ME, Window NT 4.0, Window 2000。
2. 開發工具：Visual Basic 6.0 及 DirectX 8.1。
3. 資料庫：Microsoft SQL Server 2000。

- **操作流程**

本研究採用 Window 為作業系統，並以 Visual Basic 6.0 搭配 DirectX 8.1 作為系統的開發工具。DirectX 係由微軟公司所提供，可作為開發遊戲、多媒體的工具。本研究採用 DirectX 所提供的 API，用 3D 的方式來顯示隧道開挖三維變形相關的量測資料。

由於二測點之間的位移量相對於整個隧道的尺寸是很微小的，因此需要使用誇大尺寸差異的方式來表達，在系統雛形此節，會詳細加以說明。

本系統提供點選的功能，當使用使選取量測標計點時，可開啟其他的視窗來觀看詳細資料。另外，為了可依據使用者的個人使用偏好，來顯示隧道開挖三維變形相關的量測資料，使用者可自行修正本系統相關的參數。使用者可以變更背景顏色的設定、多種的網格格線及半透明圓球之顏色的設定，及以網格線的橫向、縱向格線數目，資料來源等等。

除此之外，為了方便使用者的操控，系統提供放大、縮小、位移、旋轉、回復預設方向等功能。

當開始執行本系統之後，會出現如圖 1 所示之操作畫面。操作書面上的工

具列上有(1)連結資料庫、(2)開啟檔案、(3)顯示參數設定、(4)回復預設方向、(5)關閉本系統等按鈕。詳細之操作功能將說明如下：



圖 1 進入本系統後的操作畫面

(1) 連結資料庫

執行本功能時，會出現如圖 2 的對話方塊，將相關的資料填入之後，按連結的按鈕，即可從資料庫中讀取隧道開挖三維變形相關的量測資料。

資料庫伺服器位置	127.0.0.1
帳號	sa
密碼	*****
資料庫名稱	3dDatabase
資料表名稱	TableData

圖 2 執行本功能連結資料庫時，所先出現對話方塊

各個欄位的意義說明如下：

- 資料庫伺服器位置：SQL Server 資料庫的位置，可輸入 IP 或 Domain Name。
- 帳號：連結資料 SQL Server 所使用的登入帳號。

- 密碼：登入帳號的密碼。
- 資料庫名稱：SQL Server 中的資料庫名稱。
- 資料表名稱：資料庫中的資料表名稱。

當所輸入的資訊皆正確無誤之後，系統便將從資料庫中查詢而來之隧道開挖三維變形相關的量測資料，並顯示在系統畫面之中。

(2) 開啟檔案

若不使用資料庫的方式來讀取隧道開挖三維變形相關的量測資料，則可使用開啟檔案方式來讀入資料，如圖 3 所示。資料檔的附加檔名為 CSV，係使用 Excel 將資料以『，』分隔的方式將資料輸出至檔案。此 CSV 檔案可用 Excel 開啟或用記事本開啟來加以修改內容。

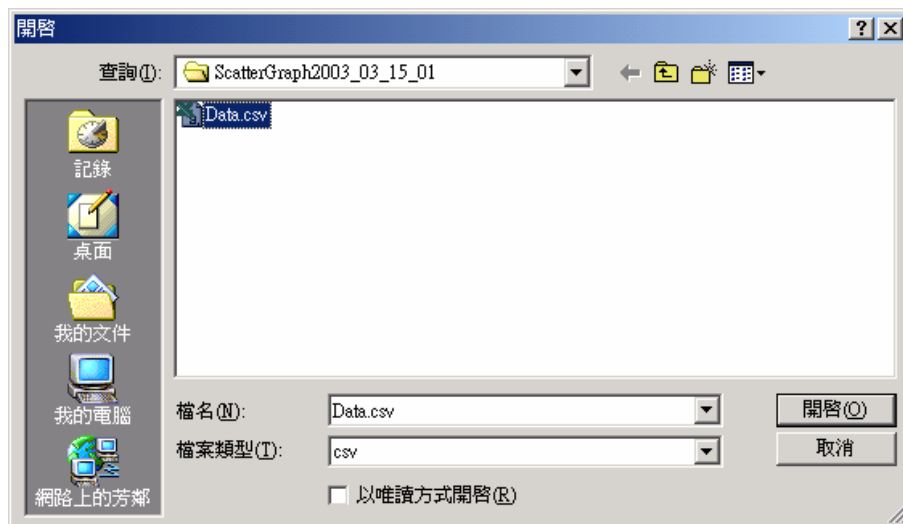


圖 3 以檔案方式讀入隧道開挖三維變形相關的量測資料

(3) 顯示系統參數設定

當使用者按了『顯示系統參數設定』按鈕之後，系統會出現如圖 4 之對話方塊，以提供使用者自行設定相關的顯示參數。

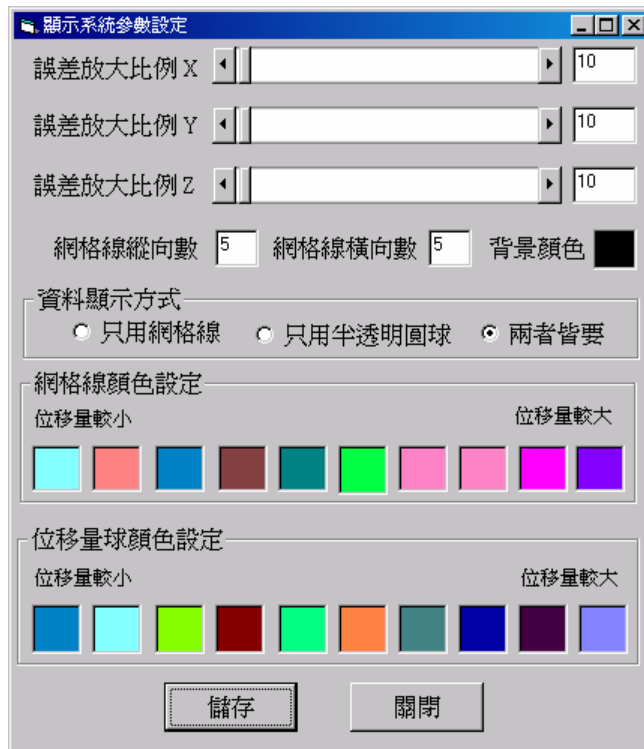


圖 4 顯示系統參數設定對話方塊

各個欄位的意義說明如下：

- 誤差放大比例 X：不同時間點量測所得之標記點在 X 軸座標值差異的放大倍率，最小值為 1 最大值為 1000，必須輸入，且為整數。
- 誤差放大比例 Y：不同時間點量測所得之標記點在 Y 軸座標值差異的放大倍率，最小值為 1 最大值為 1000，必須輸入，且為整數。
- 誤差放大比例 Z：不同時間點量測所得之標記點在 Z 軸座標值差異的放大倍率，最小值為 1 最大值為 1000，必須輸入，且為整數。
- 網格線縱向數：最小值為 5 最大值為 10，必須輸入，且為整數。
- 網格線橫向數：最小值為 5 最大值為 10，必須輸入，且為整數。
- 背景顏色：顯示系統的背景顏色。
- 資料顯示方法：隧道開挖三維變形相關顯示方式，系統提供(1)只用網格線、(2)只用半透明圓球、(3)二者皆要，等三種表示方法。
- 網格線顏色設定：依照隧道開挖三維變形的變形量，從小到大可設定 10 種顏色，使得用網格線來顯示隧道開挖三維變形資料時，可依不同的變形量來決定其顏色。
- 半透明圓球顏色設定：依照隧道開挖三維變形的變形量，從小到大可設定 10 種顏色，使得用半透明圓球來顯示隧道開挖三維變形資料時，可依不同的變形量來決定其顏色。

至於，顏色的設定方式為點選『顯示系統參數設定』對話方塊任一個顏色方格之後，會出現如圖 5 所示之色彩選擇對話方塊，供使用者自行選擇顏色。

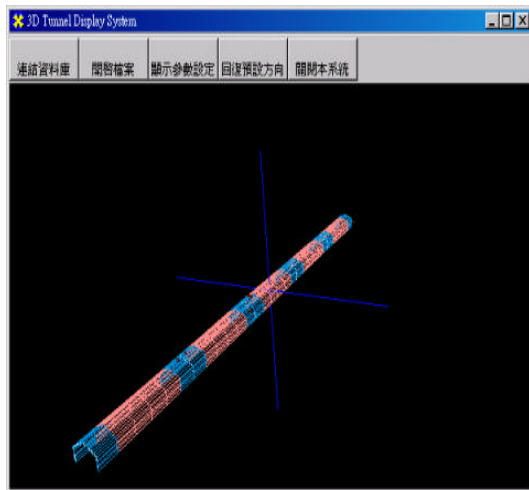


圖 5 色彩的對話方塊

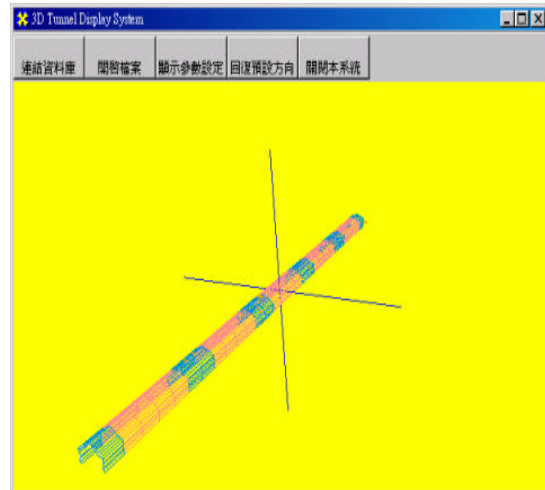
當使用者選擇某一顏色之後，顏色方格的色彩可會自動變更成使用者所選用的顏色，並套用於顯示系統中。以變更背景顏色為例，圖 6 上的背景顏色原本為黑色、可變更成黃色之後，顯示系統參數設定對話方塊的背景顏色立即變為黃色，如圖 6 所示。按顯示系統參數設定對話方塊的儲存按鈕，系統便會將參數儲存起來，並套用在顯示系統上，如圖 7 所示。



圖 6 在顯示系統參數設定對話方塊中，背景顏色已由黑色變為黃色



(1) 變更前



(2) 變更後

圖 7 背景顏色變更的結果(1)變更前(2)變更後

(4)回復預設方向

有時候顯示系統的畫面，因為使用者執行了放大、縮小、旋轉、位移等相關操作之後，畫面中的資料也會跟著變更。若使用者按了『回復預設方向』之後，所有有關於放大、縮小、旋轉、位移的操作都會被還原，而呈現系統的預設畫面。

(5)關閉本系統

當使用者不想要繼續執行時，使用者按了『關閉本系統』的按鈕之後，會出現如圖 8 的畫面，以確定使用者是否要關閉本系統。若使用者按「確定」的按鈕，則系統即關閉了；若使用者用按「取消」的按鈕，則表示使用者不想關閉本系統，所以系統可以繼續執行。

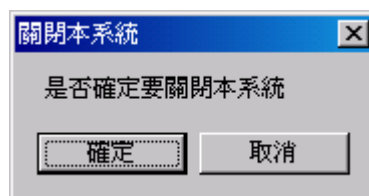
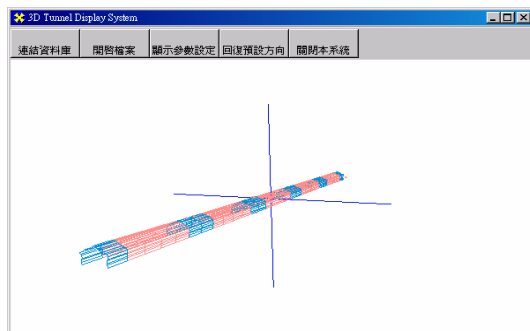


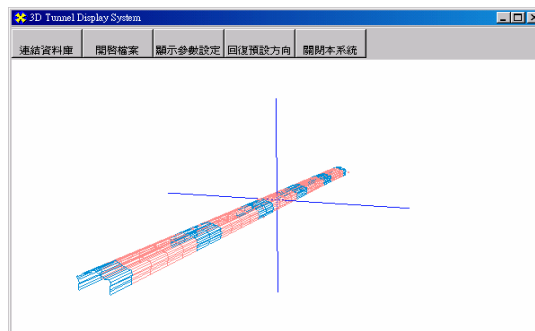
圖 8 確認使用者是否確定要關閉本系統的對話方塊

為了方便使用者的操控，除了回復預設方向的功能之外，系統提供藉由滑鼠及功能鍵作放大、縮小、位移、旋轉等功能。

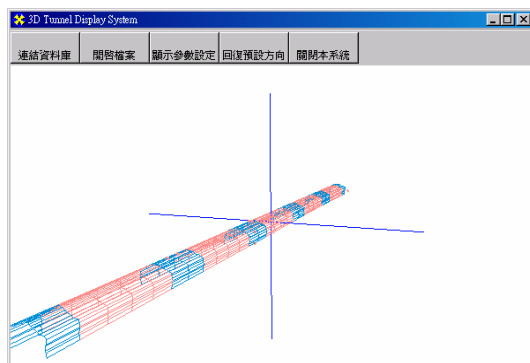
- 放大：Ctrl+w 可執行放大的功能，如圖 9 所示，為按 Ctrl+w 的放大結果。



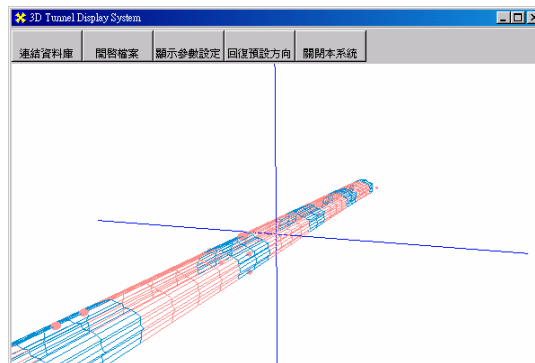
(1)原始大小



(2)放大



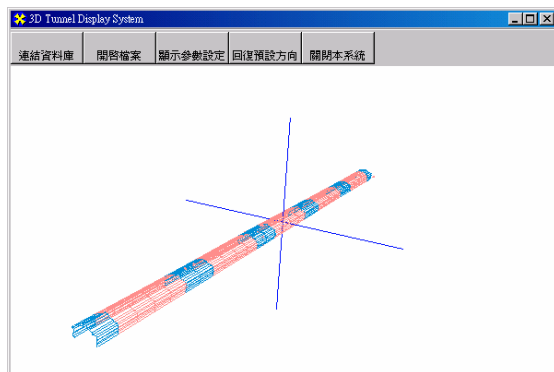
(3)放大



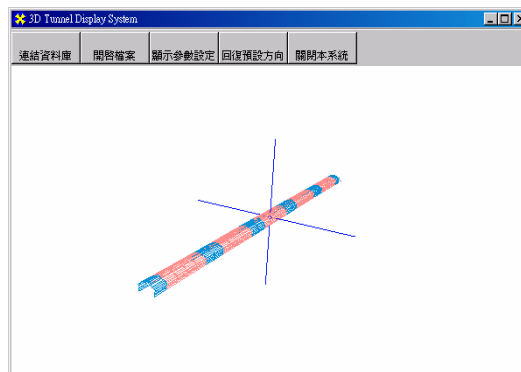
(4)放大

圖 9 按 Ctrl+w 的放大結果

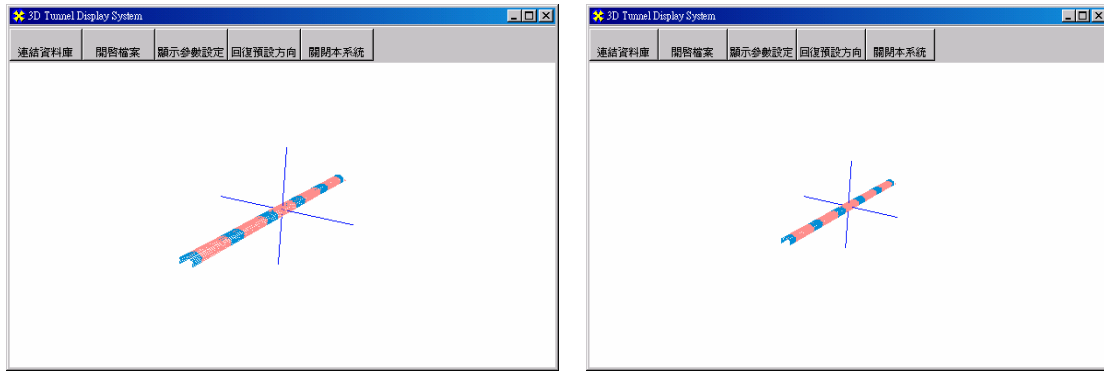
- 縮小：Ctrl+s 可執行縮小的功能，如圖 5-17 所示，為按 Ctrl+s 的縮小結果。



(1)原始大小



(2) 縮小

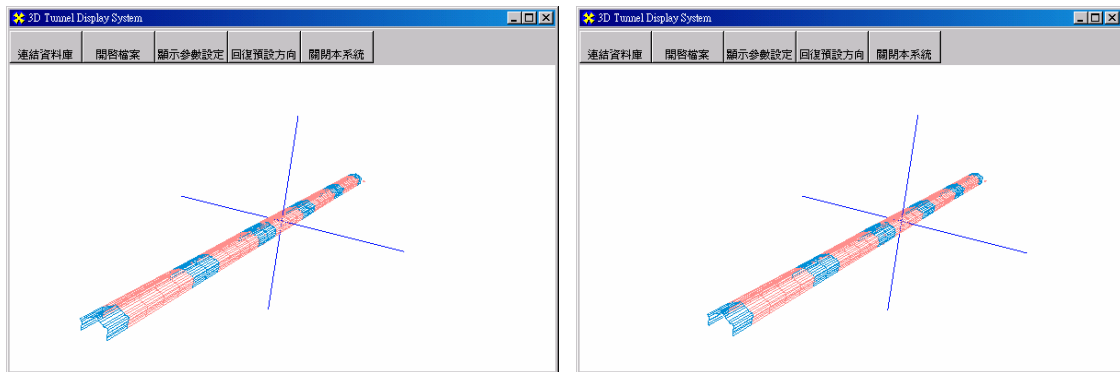


(3) 縮小

(4) 縮小

圖 10 按 Ctrl+s 的縮小結果

- 右移：Ctrl+q 可執行右移的功能，如圖 11 所示，為按 Ctrl+q 的右移結果。

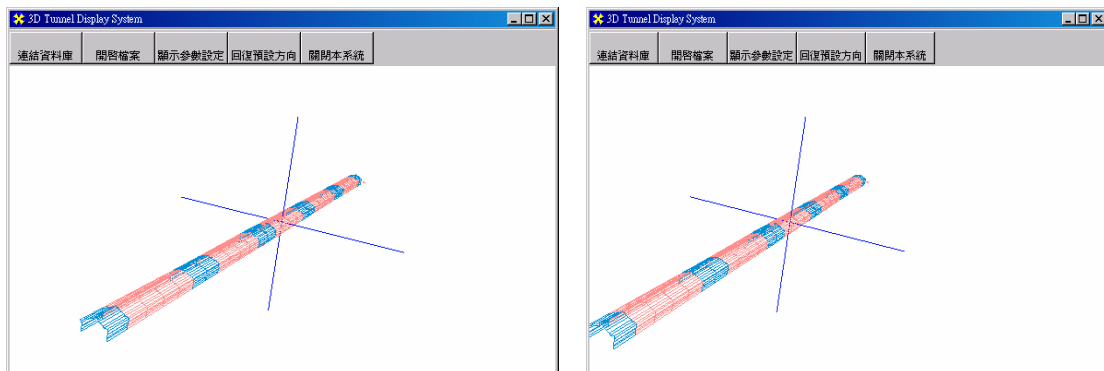


(1)原始畫面

(2)向右移動

圖 11 按 Ctrl+q 的右移結果

- 左移：Ctrl+e 可執行左移的功能，如圖 12 所示，為按 Ctrl+e 的左移結果。

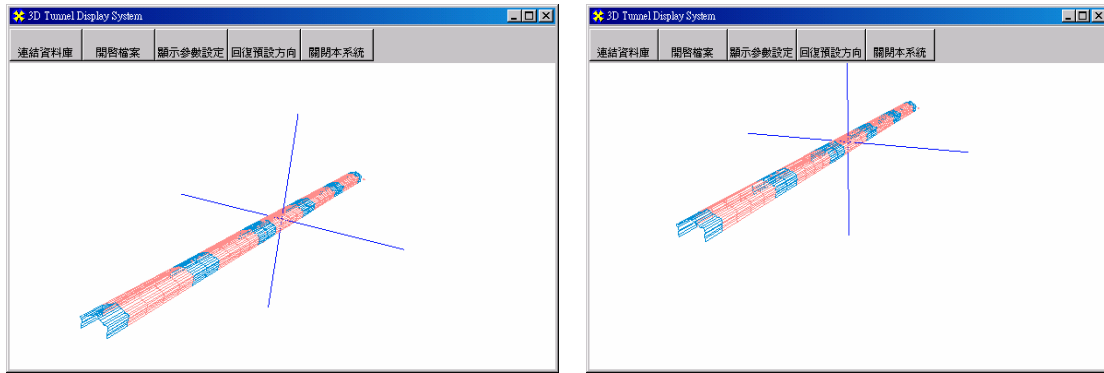


(1)原始畫面

(2)向左移動

圖 12 按 Ctrl+e 的左移結果

- 上移：Ctrl+z 可執行上移的功能，如圖 13 所示，為按 Ctrl+z 的上移結果。

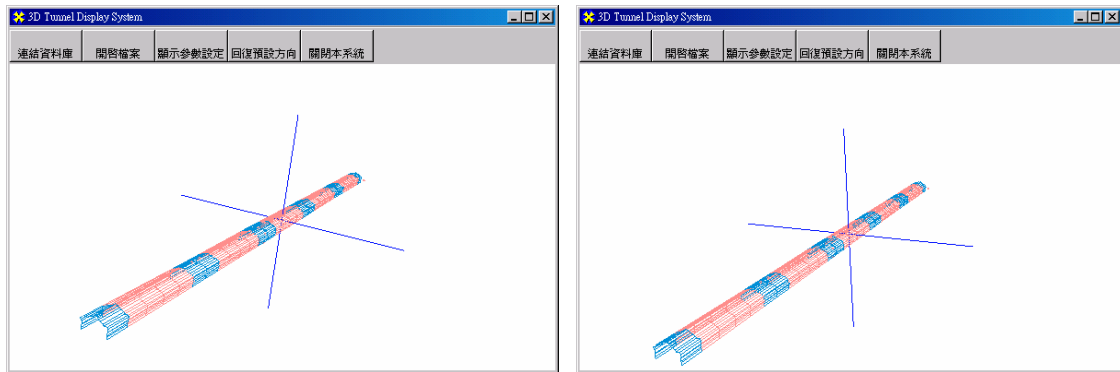


(1)原始畫面

(2)向上移動

圖 13 按 Ctrl+z 的上移結果

- 下移：Ctrl+a 可執行下移的功能，如圖 14 所示，為按 Ctrl+a 的下移結果。

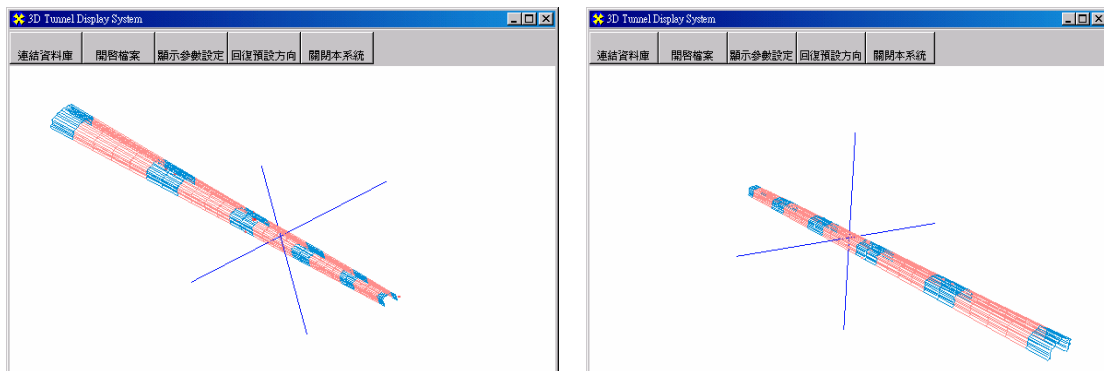


(1)原始畫面

(2)向下移動

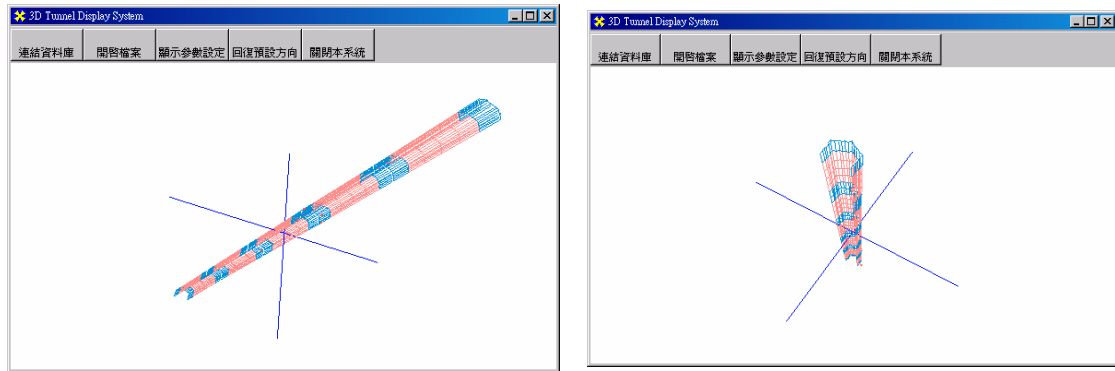
圖 14 按 Ctrl+a 的下移結果

- 旋轉：按滑鼠中鍵或左鍵不放且移動滑鼠時，系統畫面會依據滑鼠的移動方向來旋轉，圖 15 為各種各種不同的旋轉角度。



(1)

(2)



(3)

(4)

圖 15 各種不同的旋轉角度

當移動滑鼠到某一圓球上面時，畫面會出現此相對應的標記點之資料，如圖 16 所示。

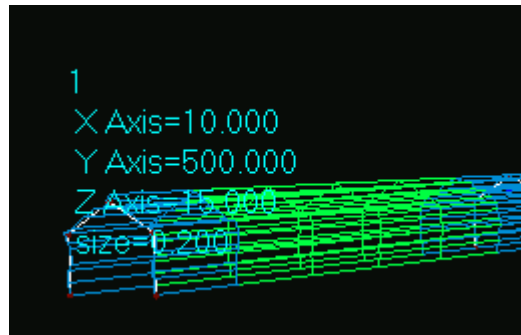


圖 16 移動滑鼠到某一圓球上面時，畫面會出現此相對應的標記點之資料

當點選某一圓球時，系統會出現一個畫面，用以顯示此相對應的標記點之資料，如圖 17 所示。



圖 17 移某一圓球時時，系統會出現一個畫面