

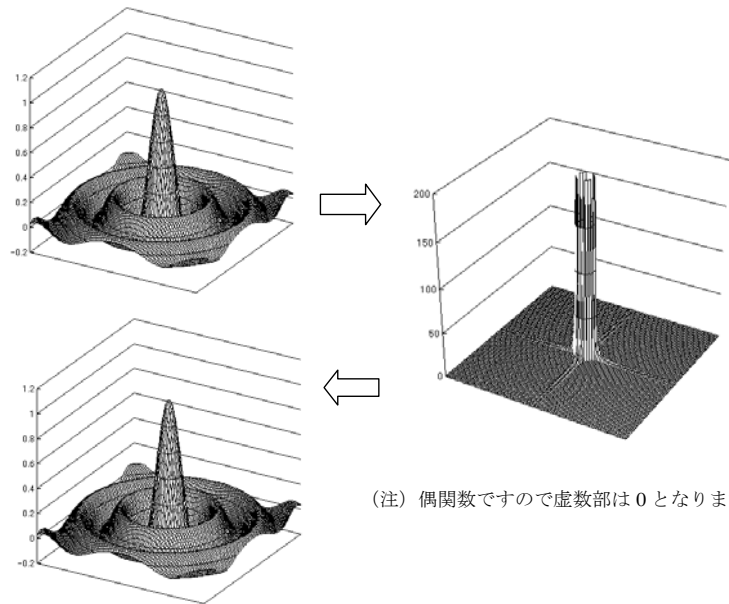
表 6-7 2次元 DFT (その3)

```

For l = 0 To m
For k = 0 To m
  Z1(l, k) = DivR(Z1(l, k), m)
Next
Next
For k = 0 To m
  For N = 0 To m
    DD = (-1) ^ (k + N)
    Z2(k, N).実部 = Z1(k, N).実部 * DD
    Z2(k, N).虚部 = Z1(k, N).虚部 * DD
  Next
Next
Next
End Sub
    
```

■結果の確認

実行結果を三次元グラフにしたのが図 6-15 です。FFT の結果を逆 FFT すると元に戻っています。



(注) 偶関数ですので虚数部は 0 となります。

図 6-15 2次元 DFT 結果

(4)写真データを DFT 用のデータに加工

写真データを赤緑青 (RGB) それぞれに分けて DFT 用のデータにすると煩雑になりますので、VB6 のピクチャボックスに読み込まれたデータを白黒のデータに変換し、DFT 用のデータ (Double 型の 100×100 の配列) に変換するプログラム例を表 6-8 に示します。

これまでのプログラムを VB6 に移行して、2次元 DFT の結果を色マップで表示した例を図 6-16 に示します。画面の上段は実数部、下段は虚数部です。また左側が元データ、中央が DFT の結果、左側が逆 DFT を行った結果です。

表 6-8 画像変換

```

Public 画像 DT(100, 100) As Double
Public Sub window(P As PictureBox, X1, Y1, X2, Y2)
  With P
    .ScaleLeft = X1: .ScaleTop = Y1
    .ScaleWidth = X2 - X1: .ScaleHeight = Y2 - Y1
  End With
End Sub
Private Sub 白黒変換()
  Dim X1 As Double: Dim C As Double
  Dim C1 As Double: Dim C2 As Double: Dim C3 As Double
  window Picture2, 0, 0, 100, 100
  W = Picture1.ScaleWidth: H = Picture1.ScaleHeight
  DW = W / 100: DH = H / 100
  For l = 0 To 100
  For J = 0 To 100
    X = DW * J: Y = DH * l: C = Picture1.Point(X, Y)
    Picture2.Line (J, l)-(J + 1, l + 1), C, BF
    If C < 0 Then
      X = 255
    Else
      C1 = (C ¥ 256) ¥ 256: C2 = (C ¥ 256) Mod 256: C3 = C Mod 256
      X = (C1 + C2 + C3) / 3
    End If
    Picture2.Line (J, l)-(J + 1, l + 1), RGB(X, X, X), BF
    画像 DT(l, J) = X
  Next
Next
Next
End Sub
    
```

[演習]

表 6-7 の関数 0 を変更して、DFT 用の元データを生成するプログラムを作成し、色々なデータの 2次元 DFT の結果を観察せよ。

(最初の例は、表示形式が異なるだけで、元データは図 6-15 に示した例と同じです)



図 6-16 色々な 2 次元 DFT の例 (その 1)

(元データの画像は複雑ですが、DFT の結果は中央に値が集まっています)

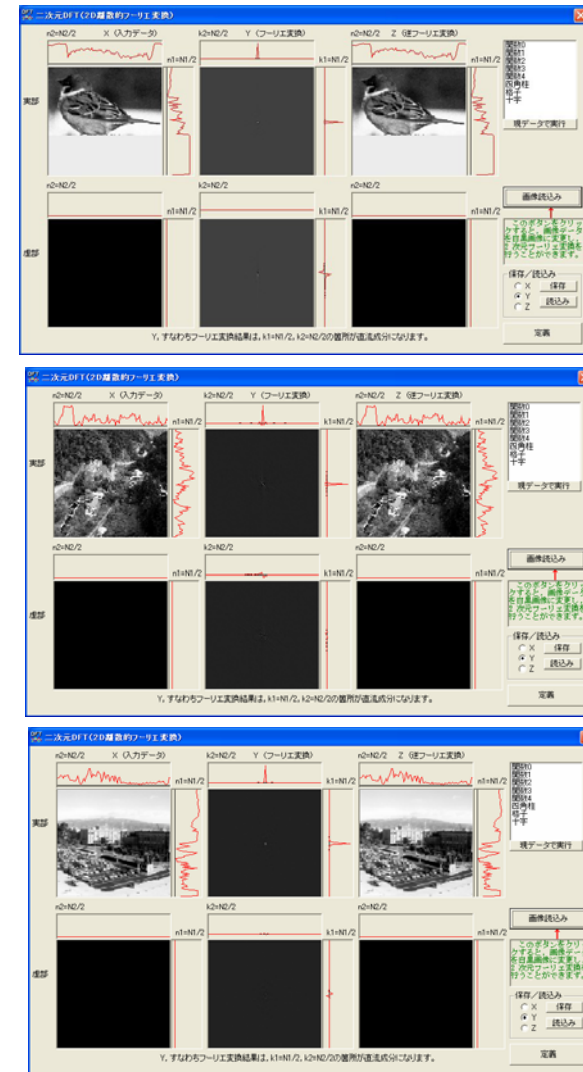


図 6-16 色々な 2 次元 DFT の例 (その 2)