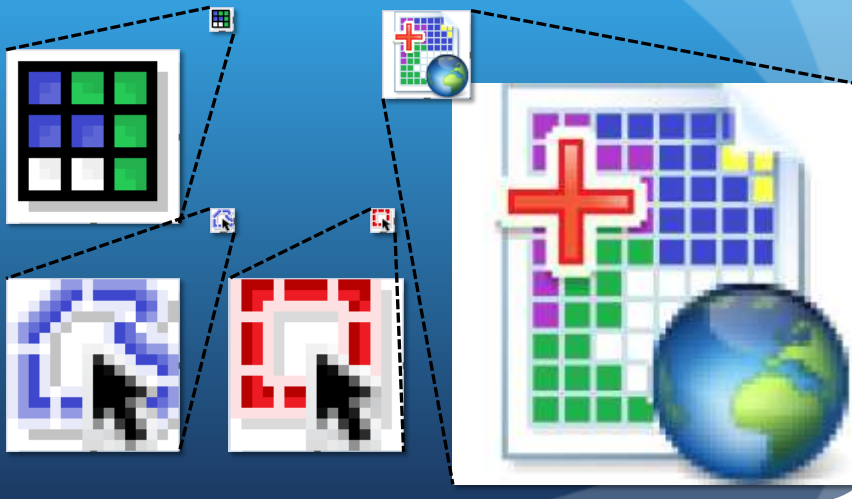


Raster Composition Calculator

ArcMap Add-In

Custom Icons



Appearance

The color associated with each pixel type is displayed in this column

This column allows users to group different pixel types together and view their cumulative percent

Raster Composition Calculator
pinierogbs

Select the raster layer to be analyzed

Launch the dockable window

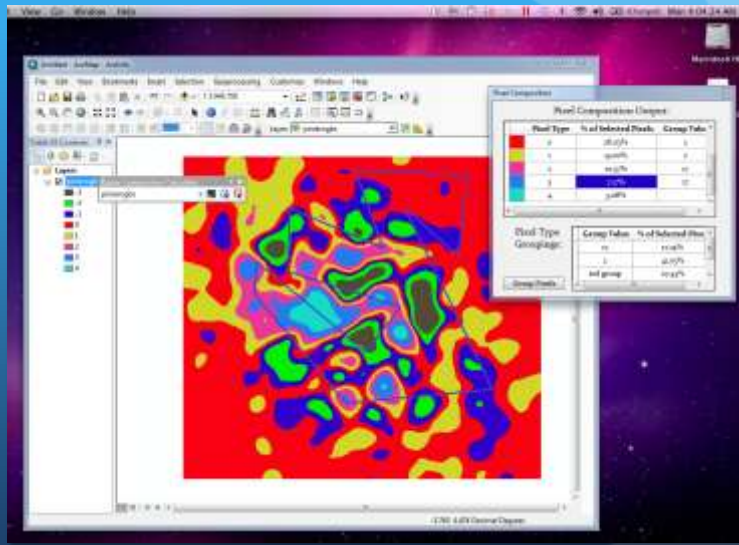
Enable the polygon and rectangle masking tools

The output for the groups that have been assigned in the upper DataGridView is displayed here. Groups can be named as number of letters/words

Pixel Type	% of Selected Pixels	Group Value
-3	6.08%	red group
-2	14.04%	blue group
-1	20.07%	1
0	12.16%	red group
1	11.82%	2

Group Value	% of Selected Pixels
1	20.07%
2	11.82%
blue group	14.04%

Demo



Obstacles

- Setting the graphic element as the mask for the raster layer
 - Initially tried to convert the graphic to feature
- Accessing the components of the add-in out side of their class
- Accessing the a pixel types color
 - Raster Unique Value Renderer
 - Delphi to Hexadecimal to RGB
- Populating and formatting DataGridViews

Snippets

mask with graphic element

```
' Find the layer in the document by matching it's name to the name in the ComboBox
For i = 0 To layerCounter
  If pMap.Layer(i).Name = rasterLayerName Then
    pRLayer = TryCast(pMap.Layer(i), RasterLayer)
  End If
Next

' Instantiate pValueRaster and pMaskPoly here
' The following six lines of code were written by Geoffrey Duh
Dim pValueRaster As IGeoDataset = pRLayer.Raster
Dim pExtractOp As IExtractionOp = New RasterExtractionOp
Dim pTempRaster As IGeoDataset = pExtractOp.Rectangle(pValueRaster, pEnv, True)

' Open attribute table
Dim pBandCol As IRasterBandCollection = pTempRaster
Dim pRasterBand As IRasterBand = pBandCol.Item(0)
Dim pTable As ITable = pRasterBand.AttributeTable

' If there are no pixels within the graphic element pTable is Nothing.
' Alert the user of this fact
If pTable Is Nothing Then
  MessageBox.Show("The graphic element does not contain any pixels.")
  Exit Sub
End If

' Instantiate a cursor and begin to cycle through the attribute table
' The following three lines of code were written by Geoffrey Duh
Dim pCur As ICursor = pTable.Search(Nothing, False)
Dim pRow As IRow = New Row
pRow = pCur.NextRow()
```

Snippets

access foreign add-in elements

```
Public Class AddInImpl
    Inherits ESRI.ArcGIS.Desktop.AddIns.DockableWindow

    Private m_windowUI As FrmMasterCompCalc

    ' The following snippet (the remaining lines of code in this class) was written
    ' by Paul Barter and was found in the following URL:
    ' http://forums.arcgis.com/threads/18507-Access-Desktop-from-Dockable-Window-from-another-class
    Friend ReadOnly Property UI() As FrmMasterCompCalc
        Get
            Return m_windowUI
        End Get
    End Property

    Protected Overrides Function OnCreateChild() As System.IntPtr
        m_windowUI = New FrmMasterCompCalc(Me, Hook)
        Return m_windowUI.Handle
    End Function

    Protected Overrides Sub Dispose(ByVal Param As Boolean)
        If m_windowUI IsNot Nothing Then
            m_windowUI.Dispose(Param)
        End If
    End Sub

    MyBase.Dispose(Param)
End Sub

Public Sub PopulateDataGridView()
    ' Enable access to the add-in's dockable window and its IDataGridView
    ' This following three lines of code were written by Paul Barter and found at the following URL:
    ' http://forums.arcgis.com/threads/18507-Access-Desktop-from-Dockable-Window-from-another-class
    Dim addInImpl As FrmMasterCompCalc.AddInImpl =
    ESRI.ArcGIS.Desktop.AddIns.AddIn.FrmID(Of FrmMasterCompCalc, AddInImpl)(My_IDs.FrmMasterCompCalc)
    Dim Window As FrmMasterCompCalc = addInImpl.UI
    Dim pTable As DataGrid

```

Snippets

```
Try
    ' Populate a list that contains all of the group values entered by the user
    ' Compare each new entry to the existing entries to ensure no repeats
    For i As Integer = 0 To DOWPiaOutput.Rows.Count - 1
        Dim bool As Boolean = False

        If groupValueList.Count > 0 Then
            For j As Integer = 0 To groupValueList.Count - 1
                If DOWPiaOutput.Rows(i).Cells(3).Value = groupValueList(j) Then
                    bool = True
                End If
            Next
        End If

        Dim tempString As String = DOWPiaOutput.Rows(i).Cells(3).Value

        If bool = False And tempString <> "" Then
            groupValueList.Add(tempString)
        End If
    Next

```

```

' Take each entry from the group list and find each entry that has that group value
' Add the pixel percentage for each entry that has the same value together
If groupValueList.Count > 0 Then
    For k As Integer = 0 To groupValueList.Count - 1
        Dim tempPixelPercent As Double = 0

        For i As Integer = 0 To DOWPiaOutput.Rows.Count - 1
            If DOWPiaOutput.Rows(i).Cells(3).Value = groupValueList(k) Then
                Dim tempString As String = DOWPiaOutput.Rows(i).Cells(2).Value

                tempString = tempString.Replace(tempString.Length - 1,
                tempPixelPercent & " %")
            End If
        Next

        groupPixelPercent.Add(tempPixelPercent)
    Next

    ' If no group values have been assigned inform the user
Else
    MessageBox.Show("No pixel types have been assigned group values.")
End Sub
End If

```

get values from datagridview1,
populate datagridview2



Questions

?