

```

1
2 //File : \Lab01_AdrianBaira\BaseBitmapManip.cs : Last Write 28-Feb-25 8:45:49 AM
3 //*****
4 //Program: LAB01_ImagePress
5 //Description: CLASS BaseBitmap
6 //Date: Feb. 24, 2025
7 //Author: Adrian Baira
8 //Course: CMPE2800
9 //Class: CNTA03
10 //*****
11 using System;
12 using System.Collections.Generic;
13 using System.Drawing;
14 using System.Linq;
15 using System.Text;
16 using System.Threading.Tasks;
17
18 namespace Lab01_AdrianBaira
19 {
20     public abstract class BaseBitmapManip
21     {
22         #region Properties
23         // a COPY of the original bitmap (assigned from the constructor)
24         public Bitmap BitmapOriginal { get; private set; }
25         #endregion
26
27         #region CTOR
28         // constructor requires a source Bitmap, and an error handler
29         // the error handler will be used for error notification
30         // when operating the provided bitmap (load, NULL, etc.)
31         public BaseBitmapManip(Bitmap bm, Action<string> error)
32         {
33             try
34             {
35                 // other error handling here?
36                 // save a copy of the image
37                 BitmapOriginal = new Bitmap(bm);
38             }
39             catch (Exception ex)
40             {
41                 error?.Invoke(ex.Message);
42             }
43         }
44         #endregion

```

70

Mar 3

1 late, no checkoff. (-30%)

Good

#region Methods

```
// build of a dictionary of colors and their frequencies from
// the source image
// this will tell us the total number of colours in the image
// and the results may be ordered to find the most popular
// colour (something of value for the manipulation)
/// <summary>
/// Building a dictionary of the Color and frequency of how much that color shows up
/// </summary>
/// <returns></returns>
```

```
public Dictionary<Color, int> BuildColourTable()
{
    Dictionary<Color, int> result = new Dictionary<Color, int>();
    // details omitted - you need to implement this
    for(int x = 0; x < BitmapOriginal.Width; x++)
    {
        for (int y = 0; y < BitmapOriginal.Height; y++)
        {
            Color color = BitmapOriginal.GetPixel(x, y);
            if(!result.ContainsKey(color))
                result.Add(color, 1);
            else
                result[color]++;
        }
    }
    return result;
}
```

```
// generate a 'difference' value from two colours
// will be compared against a threshold value
public static int GetColourDifference(Color A, Color B)
{
    int iR = Math.Abs(A.R - B.R);
    int iG = Math.Abs(A.G - B.G);
    int iB = Math.Abs(A.B - B.B);

    return iR + iG + iB;
}
```

```
// abstract method to perform the image manipulation (reduction)
// returns a new image that is the reduced version (original unmodified)
// your derived class will implement this behaviour
```

```

89         public abstract Bitmap ReduceImage(int Threshold);
90         #endregion
91     }
92
93 }
94
95 //File : \Lab01_AdrianBaira\Form1.cs : Last Write 28-Feb-25 8:45:49 AM
96 //*****
97 //Program:      LAB01_ImagePress
98 //Description:   Reducing an image
99 //Date:         Feb. 24, 2025
100 //Author:       Adrian Baira
101 //Course:       CMPE2800
102 //Class:        CNTA03
103 //*****
104 using System;
105 using System.Collections.Generic;
106 using System.ComponentModel;
107 using System.Data;
108 using System.Drawing;
109 using System.Linq;
110 using System.Text;
111 using System.Threading;
112 using System.Threading.Tasks;
113 using System.Windows.Forms;
114 using System.IO;
115
116 using static System.Net.Mime.MediaTypeNames;
117 using System.Diagnostics;
118
119 namespace Lab01_AdrianBaira
120 {
121     public partial class ImagePress : Form
122     {
123         //ThreashHold Value for what image reduces
124         int threshHoldValue = 1;
125
126         #region CTOR
127         /// <summary>
128         /// Form Constructor
129         /// Allow drop into the picture box
130         /// Subscribe to the mouse wheel event to change threshold value
131         /// </summary>
132         public ImagePress()

```

```

133 {
134     InitializeComponent();
135     UI_PictureBox.AllowDrop = true;
136     UI_TXT_ThreshHoldValue.MouseWheel += UI_TXT_ThreshHoldValue_MouseWheel;
137 }
138 #endregion
139 #region Methods
140 /// <summary>
141 /// When user scrolls it changes the value of the threshold value
142 /// and adds it to the text box
143 /// </summary>
144 /// <param name="e"> Mouse event</param>
145 private void UI_TXT_ThreshHoldValue_MouseWheel(object sender, MouseEventArgs e)
146 {
147     //when they scroll up then change value by one but stop when value reaches 256
148     if(e.Delta > 0)
149     {
150         if(threshHoldValue++ >= 256)
151             threshHoldValue = 256;
152     }
153     //scroll down stop when value is less than 1 and keep it at 1
154     else if (e.Delta < 0)
155     {
156         if(threshHoldValue-- <= 1)
157             threshHoldValue = 1;
158     }
159     UI_TXT_ThreshHoldValue.Text = threshHoldValue.ToString();
160 }
161 /// <summary>
162 /// Drag drop feature to allow user to drop a image into the picture box
163 /// </summary>
164 /// <param name="e"> Drag event</param>
165 private void UI_PictureBox_DragDrop(object sender, DragEventArgs e)
166 {
167     try
168     {
169         string[] filepath = (string[])e.Data.GetData(DataFormats.FileDrop, false);
170
171         if (filepath != null && filepath.Length > 0)
172         {
173             Bitmap bitmap = new Bitmap(filepath[0]);
174             //set the picture box to the image
175             UI_PictureBox.Image = bitmap;
176             BaseBitmapManip image = new Reduce(bitmap, x => MessageBox.Show(x));

```

```

177         UI_LBL_DragDropImagein.Text = $"There are {image.BuildColourTable().Count()} colours in this image!";
178     }
179     else
180         UI_LBL_DragDropImagein.Text = "Invalid";
181 }
182 catch (Exception ex)
183 {
184     UI_LBL_DragDropImagein.Text = ex.Message;
185 }
186 }
187 /// <summary>
188 /// When user enters the picture box to allow drag and drop to work
189 /// </summary>
190 /// <param name="sender"></param>
191 /// <param name="e"></param>
192 private void UI_PictureBox_DragEnter(object sender, DragEventArgs e)
193 {
194     if (e.Data.GetDataPresent(DataFormats.FileDrop))
195         e.Effect = DragDropEffects.Copy;
196     else
197         e.Effect = DragDropEffects.None;
198 }
199 /// <summary>
200 /// Reduce image accoring to the threshold value
201 /// </summary>
202 /// <param name="sender"></param>
203 /// <param name="e"></param>
204 private void UI_BTN_Reduce_Click(object sender, EventArgs e)
205 {
206     //IF the image in the picture ovx is empty then return no need to process
207     if (UI_PictureBox.Image == null)
208         return;
209     //Prevent user clicking reduce button if clicked twice to prevent errors
210     UI_BTN_Reduce.Enabled = false;
211
212     //making a new thread and reducing the image
213     Thread thread = new Thread(() =>
214     {
215         Bitmap bitmap = new Bitmap(UI_PictureBox.Image);
216         Reduce reducedimg = new Reduce(bitmap, Error);
217         UI_PictureBox.Image = reducedimg.ReduceImage(threshHoldValue);
218         //allowing to update the UI within the thread
219         Action updateUI = () =>
220         {

```

```

221         UI_LBL_DragDropImagein.Text = $"Reduced : {reducedimg.ColRemoved} colors in
           {reducedimg.timer.ElapsedMilliseconds} ms";
222         UI_BTN_Reduce.Enabled = true;
223     };
224
225     // Invoke the Action on the UI thread
226     this.Invoke(updateUI);
227 });
228 //start the thread and set it as a background thread
229 thread.IsBackground = true;
230 thread.Start();
231
232 }
233 /// <summary>
234 /// Error Message if the Reduce method doesnt work
235 /// </summary>
236 /// <param name="error"></param>
237 public void Error(string error)
238 {
239     MessageBox.Show(error);
240 }
241 #endregion
242 }
243 }
244
245 //File : \Lab01_AdrianBaira\Reduce.cs : Last Write 14-Mar-25 2:23:45 PM
246 //*****
247 //Program:      LAB01_ImagePress
248 //Description:  Reduce CLASS
249 //Date:         Feb. 24, 2025
250 //Author:       Adrian Baira
251 //Course:       CMPE2800
252 //Class:        CNTA03
253 //*****
254 using System;
255 using System.Collections.Generic;
256 using System.Diagnostics;
257 using System.Drawing;
258 using System.Linq;
259 using System.Text;
260 using System.Threading.Tasks;
261
262 namespace Lab01_AdrianBaira
263 {

```

```

264 public class Reduce : BaseBitmapManip
265 {
266     #region Properties
267     public Stopwatch timer { get; set; } //Timer to get the time for how long the thread took
268     public int ColRemoved { get; private set; } //Count of how many colors has it reduced to
269     #endregion
270
271     #region CTOR
272     /// <summary>
273     /// Constructor for Reducing
274     /// </summary>
275     public Reduce(Bitmap bm, Action<string> error) : base(bm, error) {}
276     #endregion
277
278     #region Overrides
279     /// <summary>
280     /// Reduces image to colors that are equal or less than threshold value and removed them from the image
281     /// </summary>
282     /// <param name="threshold">Threshold value to compare the frequency of a color in the color table</param>
283     /// <returns>A Bitmap of the new Reduced Image</returns>
284     public override Bitmap ReduceImage(int threshold)
285     {
286         //Timer for ms
287         timer = new Stopwatch();
288         timer.Start();
289         //Copy of the bitmap to change
290         Bitmap copy = new Bitmap(BitmapOriginal);
291
292         //Color table to get the Color and how many times it shows up in the picture
293         Dictionary<Color, int> colorTable = BuildColourTable();
294
295         //all the colors that are bellow or at the threshold value
296         List<Color> matchingColors = new List<Color>();
297
298         Dictionary<Color, Color> Test = new Dictionary<Color, Color>();
299
300         //loop for each Color in the Color table
301         while (colorTable.Count > 0)
302         {
303             //Always get the Color that has the biggest freq
304             var mostPop = colorTable.OrderByDescending(x => x.Value).First().Key;
305
306             //Check in the table if the color and the most popular
307             foreach (var col in colorTable.Keys)

```

✓ oh, ok.

✓

✓

} every time?
- just remove 's
order stays

```

308     {
309         if (GetColourDifference(mostPop, col) <= threshold)
310         {
311             matchingColors.Add(col);
312             Test[col] = mostPop;
313         }
314     }
315 }
316 //Testing
317 Console.WriteLine("now reduce");
318
319
320 //once done them remove all the matching colors in the color table
321 foreach (Color c in matchingColors)
322     colorTable.Remove(c);
323
324 //reset the color table
325 matchingColors.Clear();
326 ColRemoved++;
327 }
328 //to the most popular color
329 //get the pixel from the bitmap and if the pixel is in the list of matching colors then set the pixel
330 for (int x = 0; x < copy.Width; x++)
331 {
332     for (int y = 0; y < copy.Height; y++)
333     {
334         Color pixel = copy.GetPixel(x, y);
335         if (Test.ContainsKey(pixel))
336             copy.SetPixel(x, y, Test[pixel]);
337     }
338 }
339 //stop the timer
340 timer.Stop();
341 return copy;
342 }
343 #endregion
344 }
345 }
346
347 //-----Git Log-----
348 /*commit 181d80d23746ef3cc19069315d847c6db5ff732c
349 Author: Adrian Baira <144168459+ABaira@users.noreply.github.com>
350 Date: Mon Mar 3 11:51:45 2025 -0700
351

```



```
352     a
353
354 labs/Lab01_AdrianBaira/Lab01_AdrianBaira/Reduce.cs
355
356 commit d6c39f9cdac0d0c17805b91ee510b02811acddb7
357 Author: Adrian Baira <abaira1@nait.ca>
358 Date:   Mon Feb 24 19:26:07 2025 -0500
359
360     Commit i know it works with small jupiter
361
362 labs/Lab01_AdrianBaira/Images/jupiter_small.png
363
364 commit e55ec0fcdc3ff875caedef69eb0c37fa96c8dbf7
365 Author: Adrian Baira <abaira1@nait.ca>
366 Date:   Mon Feb 24 19:07:26 2025 -0500
367
368     Added more comments
369
370 labs/Lab01_AdrianBaira/Lab01_AdrianBaira/BaseBitmapManip.cs
371 labs/Lab01_AdrianBaira/Lab01_AdrianBaira/Form1.cs
372 labs/Lab01_AdrianBaira/Lab01_AdrianBaira/Reduce.cs
373
374 commit 96ca6ab40b70cbe8b205c460269753a8eab1e91c
375 Author: Adrian Baira <abaira1@nait.ca>
376 Date:   Mon Feb 24 17:29:41 2025 -0500
377
378     Added comments still testing functionality i have the green working properly just need to check the Jupiter now
379
380 labs/Lab01_AdrianBaira/Lab01_AdrianBaira/BaseBitmapManip.cs
381 labs/Lab01_AdrianBaira/Lab01_AdrianBaira/BitmapManip.cs
382 labs/Lab01_AdrianBaira/Lab01_AdrianBaira/Form1.cs
383 labs/Lab01_AdrianBaira/Lab01_AdrianBaira/Lab01_AdrianBaira.csproj
384 labs/Lab01_AdrianBaira/Lab01_AdrianBaira/Reduce.cs
385
386 commit 1e45c9fb5a2d4a9640edda244c5581d9a01fd4d0
387 Author: Adrian Baira <abaira1@nait.ca>
388 Date:   Sat Feb 22 20:27:17 2025 -0500
389
390     Commit Added reduce fuctionality
391
392 labs/Lab01_AdrianBaira/Lab01_AdrianBaira/BaseBitmapManip.cs
393 labs/Lab01_AdrianBaira/Lab01_AdrianBaira/Form1.Designer.cs
394 labs/Lab01_AdrianBaira/Lab01_AdrianBaira/Form1.cs
395 labs/Lab01_AdrianBaira/Lab01_AdrianBaira/Lab01_AdrianBaira.csproj
```

```
396 labs/Lab01_AdrianBaira/Lab01_AdrianBaira/Reduce.cs
397
398 commit 7cf019d492968d39d627403013623f16642a26d8
399 Author: Adrian Baira <abaira1@nait.ca>
400 Date: Fri Feb 21 18:41:58 2025 -0700
401
402     das
403
404 labs/Lab01_AdrianBaira/ImagePress.docx.pdf
405 labs/Lab01_AdrianBaira/Images/jupiter.png
406 labs/Lab01_AdrianBaira/Images/lingrad_green 1.bmp
407
408 commit 8df9b74cfab6ee64cfa9ac09863a8e1c8b0025da
409 Author: Adrian Baira <144168459+ABaira@users.noreply.github.com>
410 Date: Mon Feb 3 11:39:53 2025 -0700
411
412     Commit Fixed with More Comments and fixed ordering
413
414 labs/Lab01_AdrianBaira/Lab01_AdrianBaira/Saturn.jpg
415
416 commit bff08568024a7f9a1aef1f44294fad11161da8da
417 Author: Adrian Baira <abaira1@nait.ca>
418 Date: Fri Jan 31 18:23:14 2025 -0700
419
420     Fixing
421
422 labs/Lab01_AdrianBaira/Lab01_AdrianBaira/BaseBitmapManip.cs
423 labs/Lab01_AdrianBaira/Lab01_AdrianBaira/BitmapManip.cs
424 labs/Lab01_AdrianBaira/Lab01_AdrianBaira/Form1.cs
425 labs/Lab01_AdrianBaira/Lab01_AdrianBaira/Lab01_AdrianBaira.csproj
426
427 commit c1680026e8c53749a8b390e80d5b1e0061d58b30
428 Author: Adrian Baira <abaira1@nait.ca>
429 Date: Thu Jan 30 14:40:52 2025 -0700
430
431     Added Scroll wheel functionality for thresh hold value
432
433 labs/Lab01_AdrianBaira/Lab01_AdrianBaira/Form1.Designer.cs
434 labs/Lab01_AdrianBaira/Lab01_AdrianBaira/Form1.cs
435
436 commit f2b0c043e5157b4078051d7f57774ec6a21fe77c
437 Author: Adrian Baira <abaira1@nait.ca>
438 Date: Thu Jan 30 14:31:31 2025 -0700
439
```

```
440         Added Drag drop feature to the picture box
441
442     labs/Lab01_AdrianBaira/Lab01_AdrianBaira/BaseBitmapManip.cs
443     labs/Lab01_AdrianBaira/Lab01_AdrianBaira/Form1.Designer.cs
444     labs/Lab01_AdrianBaira/Lab01_AdrianBaira/Form1.cs
445     labs/Lab01_AdrianBaira/Lab01_AdrianBaira/Lab01_AdrianBaira.csproj
446
447     commit 12c309a0f51df52eb6257e13ff2e9500f33575dd
448     Author: Adrian Baira <abaira1@nait.ca>
449     Date:   Thu Jan 30 14:16:06 2025 -0700
450
451         Commit
452
453     labs/Lab01_AdrianBaira/Lab01_AdrianBaira.sln
454     labs/Lab01_AdrianBaira/Lab01_AdrianBaira/App.config
455     labs/Lab01_AdrianBaira/Lab01_AdrianBaira/BaseBitmapManip.cs
456     labs/Lab01_AdrianBaira/Lab01_AdrianBaira/Form1.Designer.cs
457     labs/Lab01_AdrianBaira/Lab01_AdrianBaira/Form1.cs
458     labs/Lab01_AdrianBaira/Lab01_AdrianBaira/Form1.resx
459     labs/Lab01_AdrianBaira/Lab01_AdrianBaira/Lab01_AdrianBaira.csproj
460     labs/Lab01_AdrianBaira/Lab01_AdrianBaira/Program.cs
461     labs/Lab01_AdrianBaira/Lab01_AdrianBaira/Properties/AssemblyInfo.cs
462     labs/Lab01_AdrianBaira/Lab01_AdrianBaira/Properties/Resources.Designer.cs
463     labs/Lab01_AdrianBaira/Lab01_AdrianBaira/Properties/Resources.resx
464     labs/Lab01_AdrianBaira/Lab01_AdrianBaira/Properties/Settings.Designer.cs
465     labs/Lab01_AdrianBaira/Lab01_AdrianBaira/Properties/Settings.settings
466     */
467
```