PIXELS TO PICTURES

A PROGRAMMING COURSE ON IMAGES WITH MATLAB

Instructor Guide

Module 15: Creating Custom Collages

Prerequisite Domain Knowledge: Reading images, resizing images, concatenation

Expected Completion Time: 50 minutes

Table of Contents

Simple Collage

Expected Duration: 25 minutes

Learning Objectives

- Apply the concept of concatenation to create a simple collage

Materials

- MATLAB®

Steps

Tell the students that we will be learning how to create collages from different images. We will be using concatenation and overlaying images on one another – all concepts they are familiar with by now, but just a different application of the same concepts to get different results.

First, we will start by creating a simple collage with different images. Creating a simple collage is very similar to creating complex color filters. We need to:

1. Read in the images we would like to use in the collage

2. Resize the images as needed
3. Concatenate them so that the end result is a square or a rectangle

Ask the students to open the following script:

```
open 'simpleCollage.mlx'
```

Create a Simple Collage

Read in images

```
I1 = imread('eiffel.jpg');
I2 = imread('colosseum.jpg');
I3 = imread('niagara.jpg');
```

Resize all images

```
I1 = imresize(I1,[600 300]);
I2 = imresize(I2,[300 400]);
I3 = imresize(I3,[300 400]);
```

Concatenate images into Collage

```
I = [I3 ; I2];
collage = [I I1];
```

Display final image

This script reads in three images, resizes them as needed and concatenates them. The students should be familiar with all the commands and functions here apart from the last line. Explain to the students that we will get to the last line in a minute and ask them to run the script.

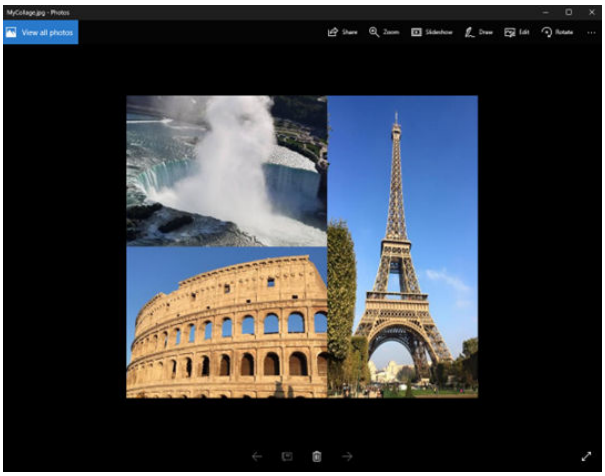
Give the students a few minutes to ensure they understand what the script is doing. Once they are comfortable, ask them what they think the last line is supposed to be doing:

```
imwrite(collage,'MyCollage.jpg')
```

The function `imwrite` saves the image variable from MATLAB into a JPEG file in the computer. It takes two inputs:

- Image - (RGB image) 3-D matrix that represents the image we want to save
- Filename - (char array) Desired name of the JPG file

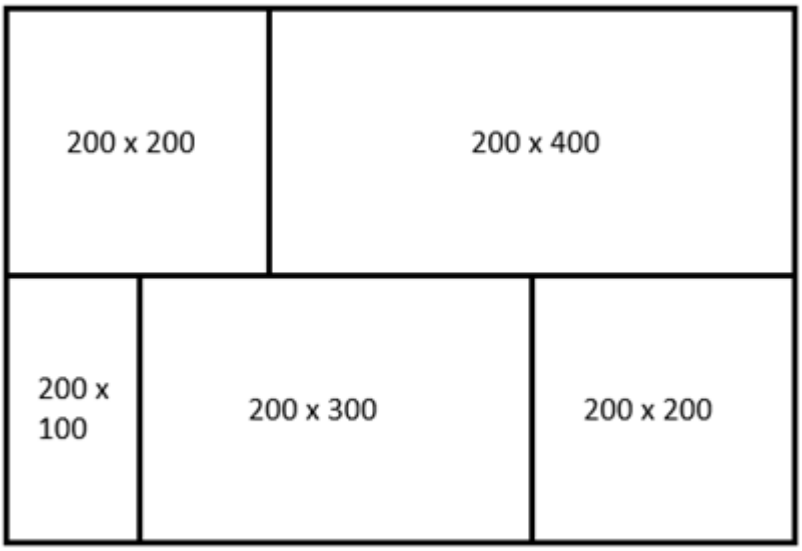
This creates a 'MyCollage.jpg' file in the Current Folder in MATLAB:



It is essentially doing that opposite of `imread` which reads an image file from the computer into MATLAB. The ability to save their image variables as JPEG files will come in very useful because they can save all the artwork they make and take it home with them.

Give the students a few minutes to create their own collages. They can edit the script to read in as many images as they like and resize them as they see fit.

Students may find it helpful to draw out their collage designs on a piece of paper first:



As long as they can ensure that the concatenation rules hold – when concatenating sideways, the number of rows must match and when concatenating vertically, the number of columns must match – the concatenation should be valid.

Advanced Collage

Expected Duration: 25 minutes

Learning Objectives

- Apply the concept of concatenation to create a simple collage

Materials

- MATLAB®
- Handout "MATLAB Functions"

Steps

In the Command Window, read in an image called ‘CollageAdvanced.jpg’ and view it:

```
I = imread('CollageAdvanced.jpg')
imshow(I)
```

Ask the students to take a break from the computer and look at the image. If they were creating this collage by hand, ask them what materials they might need. List the things out on the board as they say them.

1. A background white piece of paper
2. Several pictures to stick on the white background
3. Scissors to cut the images to small sizes
4. Tape or glue to stick the pictures on the background
5. Different colored pens to write the message

Ask the students to consult their **MATLAB Functions h**andout and see if they can find functions there that do the above things in MATLAB:

```
web('worksheets_and_handouts/MATLABFunctions.pdf', '-browser');
```

As the students suggest names of functions, write them out next to the list on the board:

A background white piece of paper

- Students may not come up with this function name immediately, but **`createNewPage`** creates a blank page with specified number of rows and columns.

Several pictures to stick on the white background

- **`imread`** can read in as many images as they want

Scissors to cut the images to small sizes

- **`imresize`** function can help them resize the input images to smaller sizes

Tape or glue to stick the pictures on the background

- The **`superimpose`** function can place any image over any other image – sort of like gluing

Different colored pens to write the message

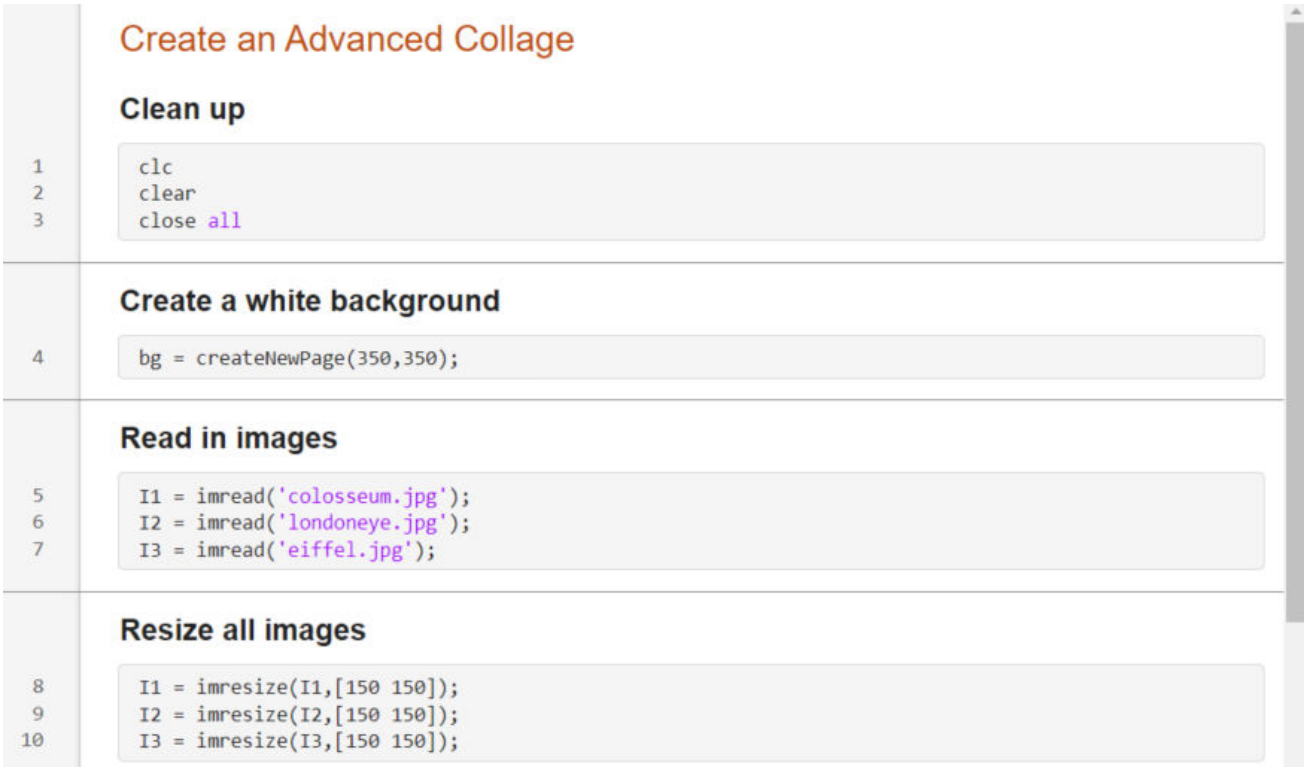
- Ask the students what function can be used to write text on an image – like a meme. The same function can be used here – **`writetext`**.

The end result on the board should look something similar to:

A background white piece of paper	**`createNewPage`**
Several pictures to stuck on the background	**`imread`**
Scissors to cut the images into smaller sizes	**`imresize`**
Tape or glue to stick the pictures on the background	**`superImpose`**
Colored pens to write the message	**`writetext`**

Ask the students to open the following script:

```
open 'advancedCollage.mlx'
```



Create an Advanced Collage

Clean up

```
clc
clear
close all
```

Create a white background

```
bg = createNewPage(350,350);
```

Read in images

```
I1 = imread('colosseum.jpg');
I2 = imread('londoneye.jpg');
I3 = imread('eiffel.jpg');
```

Resize all images

```
I1 = imresize(I1,[150 150]);
I2 = imresize(I2,[150 150]);
I3 = imresize(I3,[150 150]);
```

Verify that the sections in the script match the steps listed on the board. Run the script to ensure it matches the 'AdvancedCollage.jpg' image.

First, give them a few minutes to understand what each section of the script is doing before going through it together.

Encourage the students to modify the script as they like, they can bring in their personal images, resize them as needed and place them wherever they like on the background "paper". Remind them they can even change the color and the size of the background if they want. Also remind the students about the `getPoints` function if they need help in finding out the location of where to place the images or text in the collage.

Let the students know that if they want to save their collages as JPEG files they need to use the `imwrite` function by adding a final section at the end as shown below.

```
imwrite(collageText, 'PersonalCollage.jpg');
```