PIXELS TO PICTURES

A PROGRAMMING COURSE ON IMAGES WITH MATLAB

Instructor Guide

Module 10: Complex Color Filters

Prerequisite Domain Knowledge: Matrices, Concatenation, Functions

Expected Completion Time: 50 minutes

Table of Contents

Creating Complex Color Filters with MATLAB

Expected Duration: 30 minutes

Learning Objectives

- Apply addition and concatenation of matrices together to create complex color filters

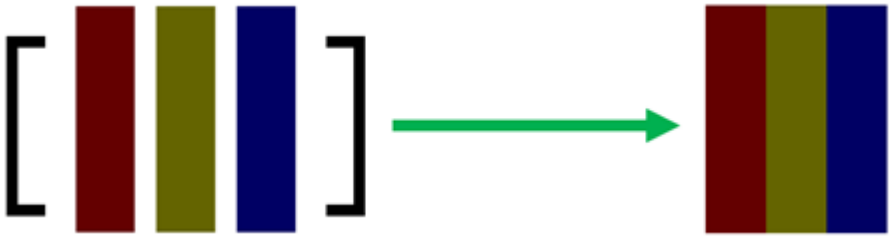
Materials

- MATLAB®

Steps

Tell the students that we will use both the concepts of addition and concatenation to create elaborate filters.

Show them the image below and explain what we will be doing in this section. We will first create thin strips of color filters and concatenate them to create one big color filter:



Then we will add this filter to any image of our choice:

Ask:

What steps will we need to carry out to complete this activity?

The students should mention:

1. Read in an image
2. Resize it to nice divisible number of rows and columns (e.g. 600 rows and 600 columns)
3. View image
4. Create 3 different colored filters with 600 rows and 200 columns each
5. Concatenate the color filters row-wise
6. Add final filter to input image
7. View final image

Ask the students to open a script called `ComplexFilters.mlx`:

```
open 'ComplexFilters.mlx'
```

This will open up a template script with the steps mentioned above, where each step is divided in to a different section. Instruct the students to run the script section-by-section using the **Run and Advance** button in the **Editor tab** (next to the **Run** button) and ensure they understand each line of code. All the commands and functions should be familiar to them at this point.

Complex Filters

Clean up

```
clc
clear
close all
```

Read in an image

```
In = imread('swan.jpg');
```

Resize input image to 600 rows and 600 columns

```
In = imresize(In,[600 600]);
```

View image

```
imshow(In)
```

Create 3 different colored filters with 600 rows and 200 columns each

```
red = newFilter(600,200,[100 0 0]);
yellow = newFilter(600,200,[100 100 0]);
blue = newFilter(600,200,[0 0 100]);
```

View the filters

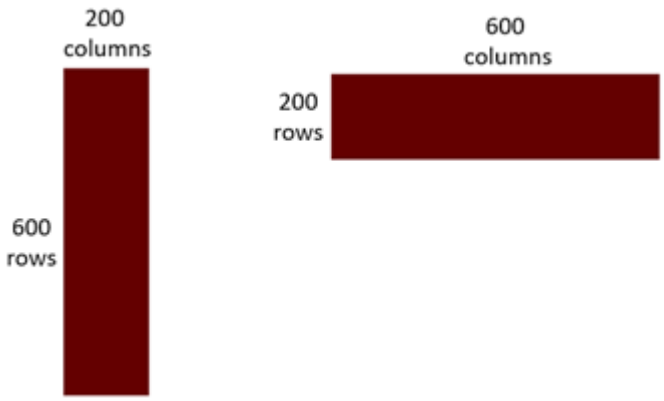
```
figure
imshow(red)
figure
```

Using this script, have the students:

- Change the colors of the three filters to get different colored complex filters.
- Change the input image to apply the complex filter to different images.

Ask the students what changes do they have to make to the script to get horizontal striped filters instead of vertical stripes.

- The size of the filters need to be changed. Instead of 600 rows and 200 columns, they need to have 200 rows and 600 columns.



- The filters need to be concatenated column-wise (vertically) instead of row-wise

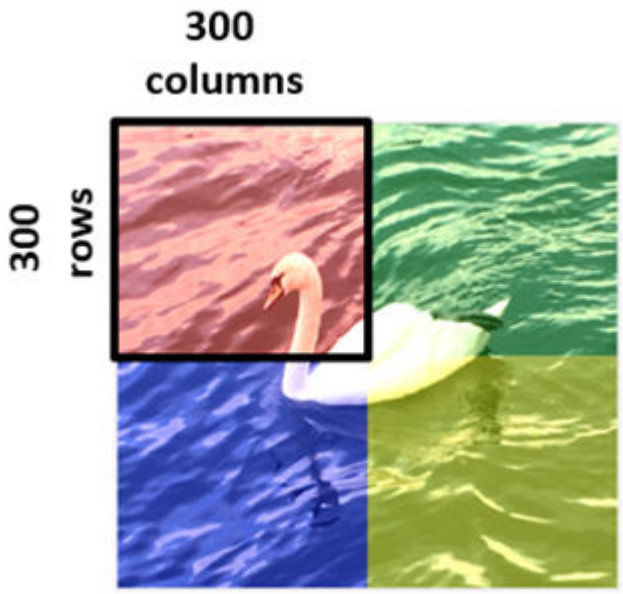
Make the following changes to the script and ask the students to follow along:

```
%% Create 3 different colored filters with 600 rows and 200 columns each
red = newFilter(200,600,[100 0 0]);
yellow = newFilter(200,600,[100 100 0]);
blue = newFilter(200,600,[0 0 100]);
```

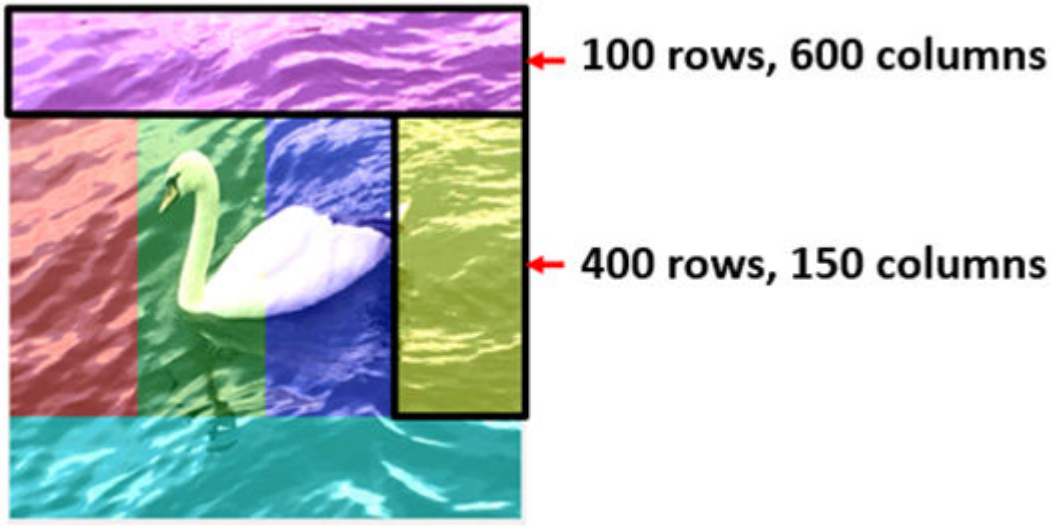
```
%% Concatenate the color filters row-wise
colors = [red ; yellow ; blue];
```

The result should now be as follows:

Ask the students how they would create the designs below and give them a few moments to try it out.




```
colors = [red green ; blue yellow];
```




```
colors = [pink ; red green blue yellow ; cyan];
```

Creating Complex Color Filters with Color Gel Sheets

Expected Duration: 20 minutes

Learning Objectives

- Create imaginative complex color filter designs as a group
- Analyze how they would concatenate it to get the same design in MATLAB

Materials

- MATLAB®
- Colored gel sheets
- White paper
- Pens and pencils
- Pairs of scissors
- Tape

Steps

Now that the students have a general idea of how to create complex color filters, tell them that they will now get a chance to take a break from the coding and do something creative!

Divide the students into groups of 3 or 4.

Spread out several different colored color gel sheets on the desk. Ask each group to pick up the following supplies for their activity:

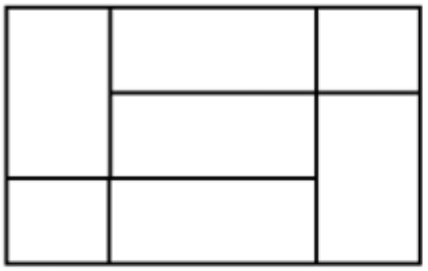
- 4 or 5 different colored gel sheets
- 1 sheet of white paper
- 1 pen or pencil
- 1 pair of scissors
- 1 sticky tape

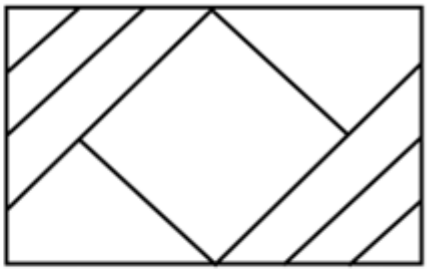
Tell the students that the goal for this activity is to come up with their unique design of a complex color filter. They can first brainstorm ideas on the white sheet of paper and then use the scissors and sticky tape to cut their color gel sheets and stitch the pieces together to create their design.

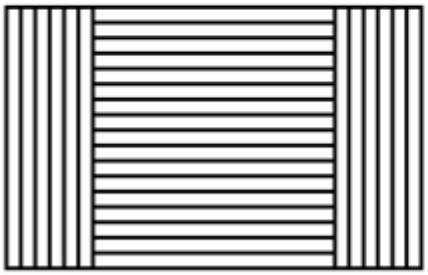
Let them know that they are welcome to create more than one design as long as they are economical in their use of the color gel sheets.

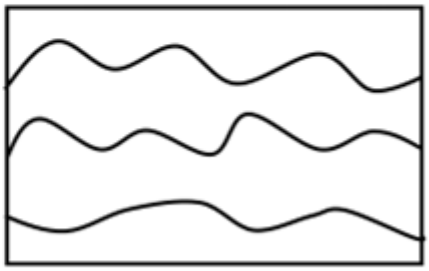
Stress on the point that they can only cut the color gel sheets into squares or rectangles (as these are the only valid matrix shapes). You can also give them a few examples of valid and invalid complex color filter designs by drawing them out on the board.

For example:









VALID

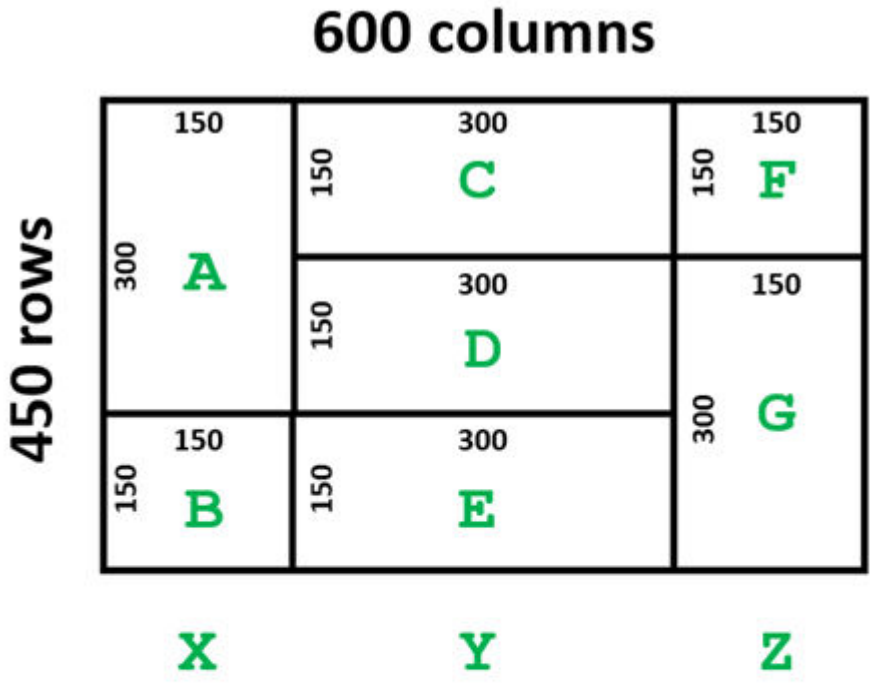
INVALID

Give the students a few minutes to work on their design. Let them know that they should get their design approved by you before they start cutting the color gel sheets. **Ensure that each group has only rectangles and squares in their design and not any other shape**. Once approved, students can then get started on implementing their designs.

Once everyone has completed, encourage the students to go around the room and look at each other's design ideas.

Still working in groups, ask the students to figure out how they would create the filter in MATLAB. Going back to the white sheet, ask them to select the dimensions of the final filter as some large values, such as 450 rows by 600 columns, and work out the dimensions of the smaller squares and rectangles from that.

For example:




```
X = [ A ; B ]
Y = [ C ; D ; E ]
Z = [ F ; G ]
Final = [ X Y Z]
```

Once the students have calculated the dimensions and figured out how they could concatenate the smaller pieces, give the students 10-15 minutes to work on their filters in MATLAB.

When time has expired, gather the students. Ask:

- *How can they apply their custom filter to an image?*
- Students should say that they can either resize the image to be the same size as their filter:

```
image = imresize(image,[450,600]);
```

or resize the filter to be the same size of the image.

```
sz = size(image);
final = imresize(final,[sz(1) sz(2)]);
```

- *What did their final image look like with the filter?*
- Students should share what their final images look like.
- *What interesting filters did you or your classmates make?*

- Accept all reasonable answers.

Encourage the students to walk around the view the work of other students