

## HW11: Image Analysis

Date Due: 11:59pm, Sun– May 10, 2015

**Problem 1:** (15 Points) Write a function with the header

```
function [newX] = myRotate90(X)
```

which rotates a matrix (such as an image) by 90 degrees. You may not use functions **rot90.m** or **imrotate.m**.

Test Case 1:

```
>> x
```

```
x =
```

0.8147	0.0975	0.1576	0.1419	0.6557
0.9058	0.2785	0.9706	0.4218	0.0357
0.1270	0.5469	0.9572	0.9157	0.8491
0.9134	0.9575	0.4854	0.7922	0.9340
0.6324	0.9649	0.8003	0.9595	0.6787

```
>> myRotate90(x)
```

```
ans =
```

0.6324	0.9134	0.1270	0.9058	0.8147
0.9649	0.9575	0.5469	0.2785	0.0975
0.8003	0.4854	0.9572	0.9706	0.1576
0.9595	0.7922	0.9157	0.4218	0.1419
0.6787	0.9340	0.8491	0.0357	0.6557

```
>>
```

**Problem 2:** (10 Points) Write a function with the header

```
function [newX] = myInvertColors(X)
```

where X is a 3-dimensional matrix containing double values between 0 and 1 where

$X(:, :, 1)$  are the red levels

$X(:, :, 2)$  are the green levels

$X(:, :, 3)$  are the blue levels

This function will switch each channel level with its complement. For example,

if  $X(10,20,1)$  had a value of 0.35, it will become 0.65

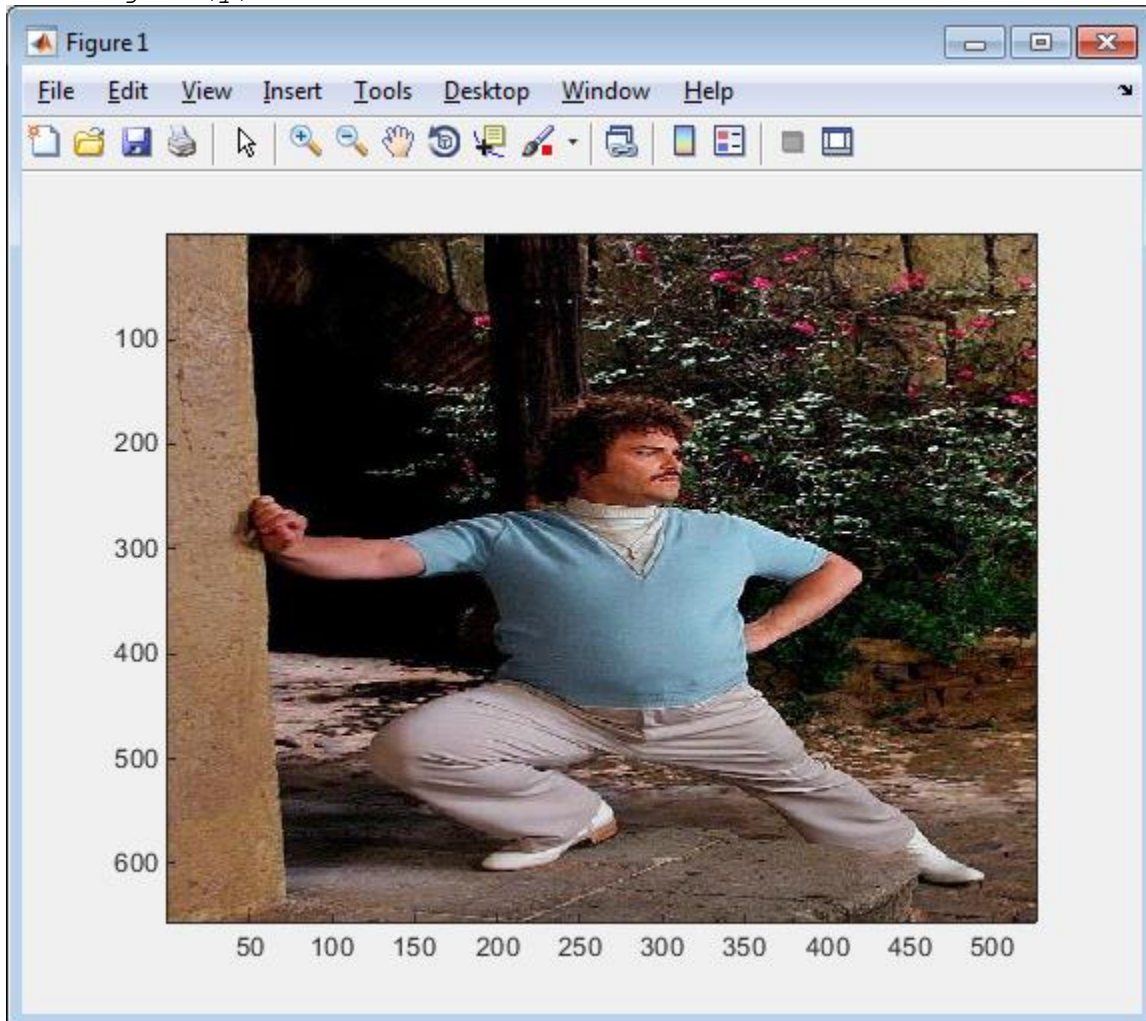
if  $X(10,20,2)$  had a value of 0.9, it will become 0.1

if  $X(10,20,3)$  had a value of 0.2, it will become 0.8

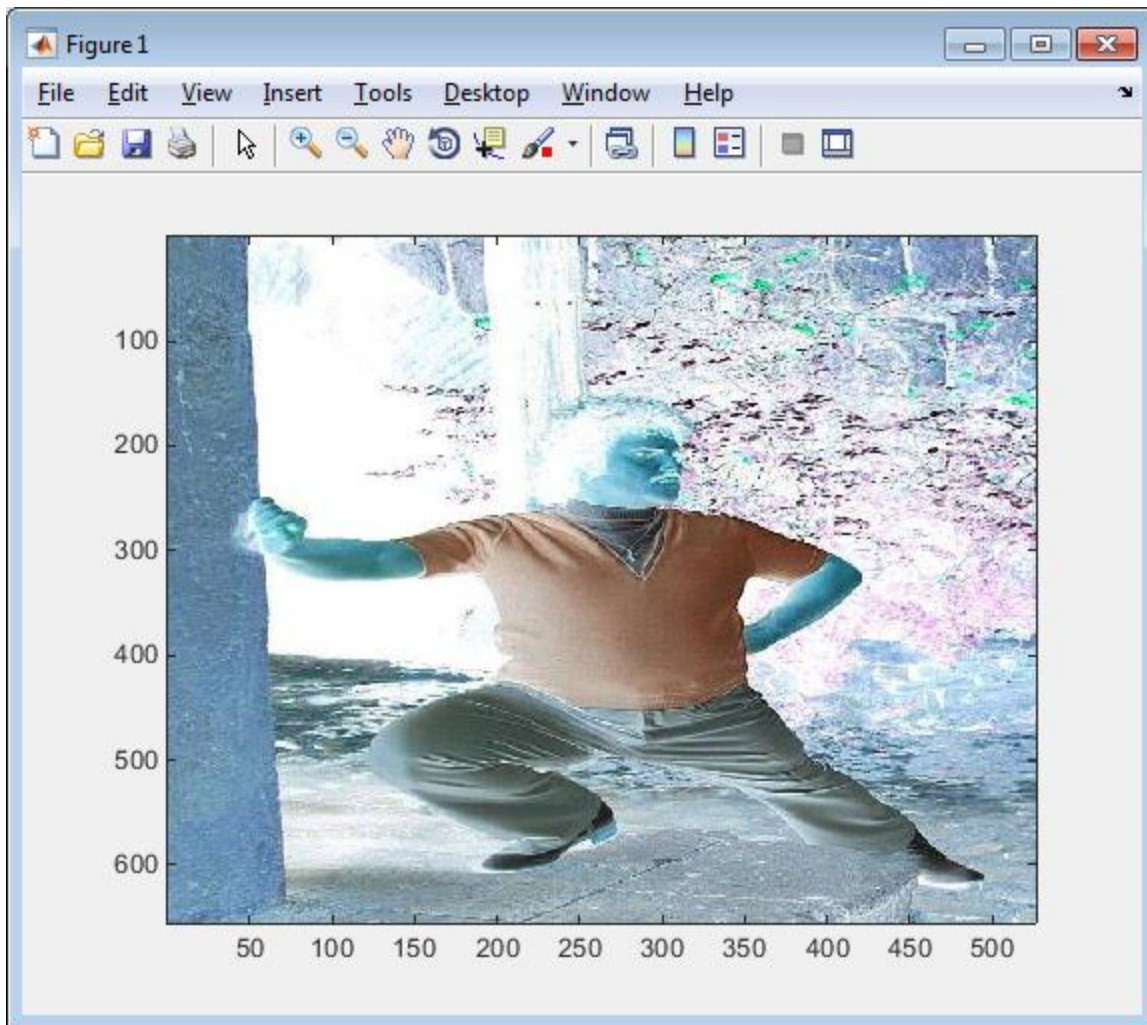
Test Case 1 (download file Test.mat)

```
>> load Test.mat
```

```
>> imagesc(y);
```



```
>> imagesc(myInvertColors(y));
```



**Problem 3:** (15 Points). Write a function with the header

```
function [newX] = myIntensity(X, rgb)
```

which takes

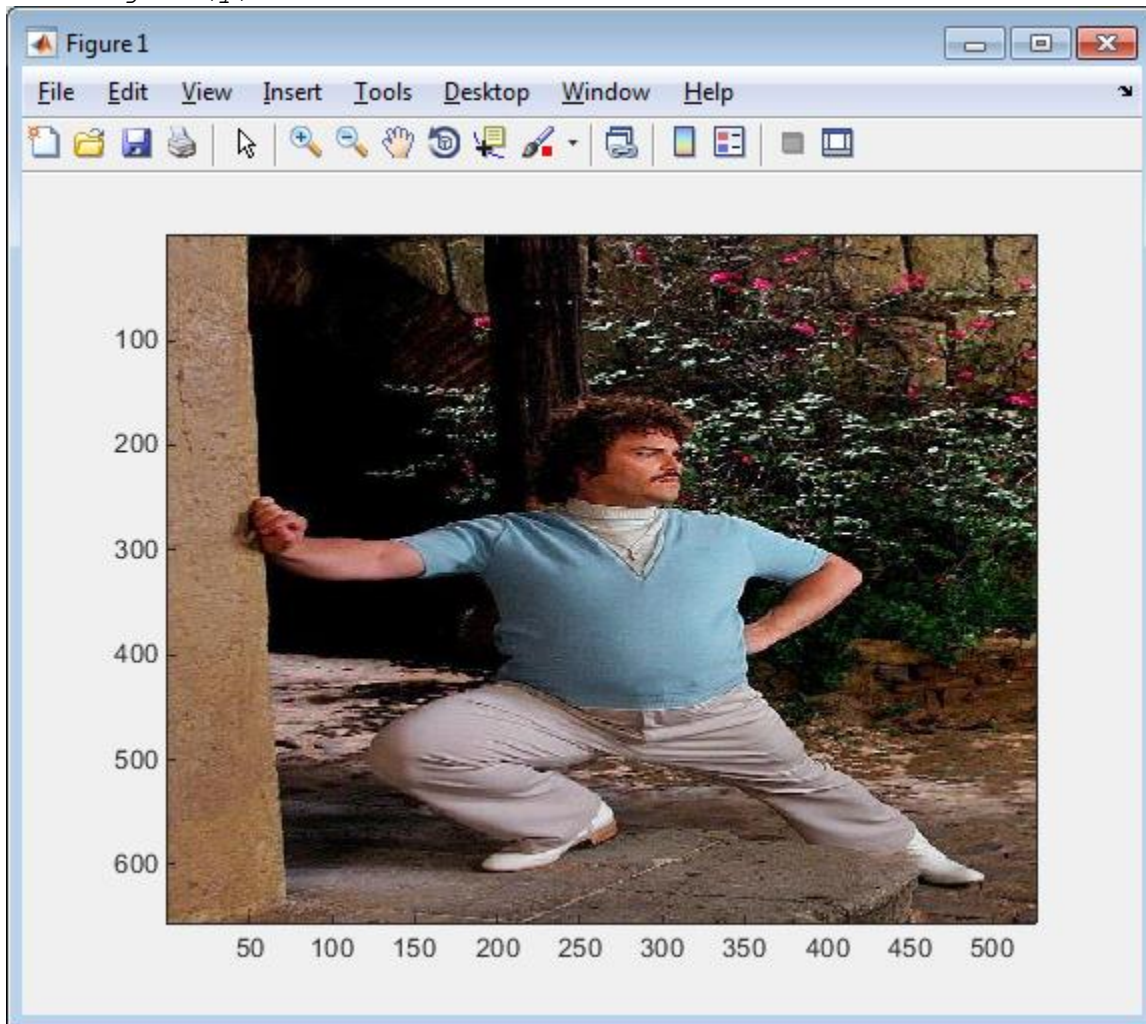
- (m,n,3) matrix X. Red components are stored in  $X(:, :, 1)$ . Green components are stored in  $X(:, :, 2)$ . Blue components are stored in  $X(:, :, 3)$ . Each component is limited to values between 0 and 1. (As in the file Test.mat).
- A 3 element vector rgb whose values are between 0 and 1.

This function allows the user to increase each color channel from its current value up to 1. To do this, I recommend the following approach:

- 1) Calculate the difference between 1 and the current red level
- 2) Calculate the difference between 1 and the current green level
- 3) Calculate the difference between 1 and the current blue level
- 4) Multiply each difference by the appropriate element of vector rgb.
- 5) Add each product from step 4 to the appropriate matrix (e.g.,  $X(:, :, 1)$  or  $X(:, :, 2)$  or  $X(:, :, 3)$ ).

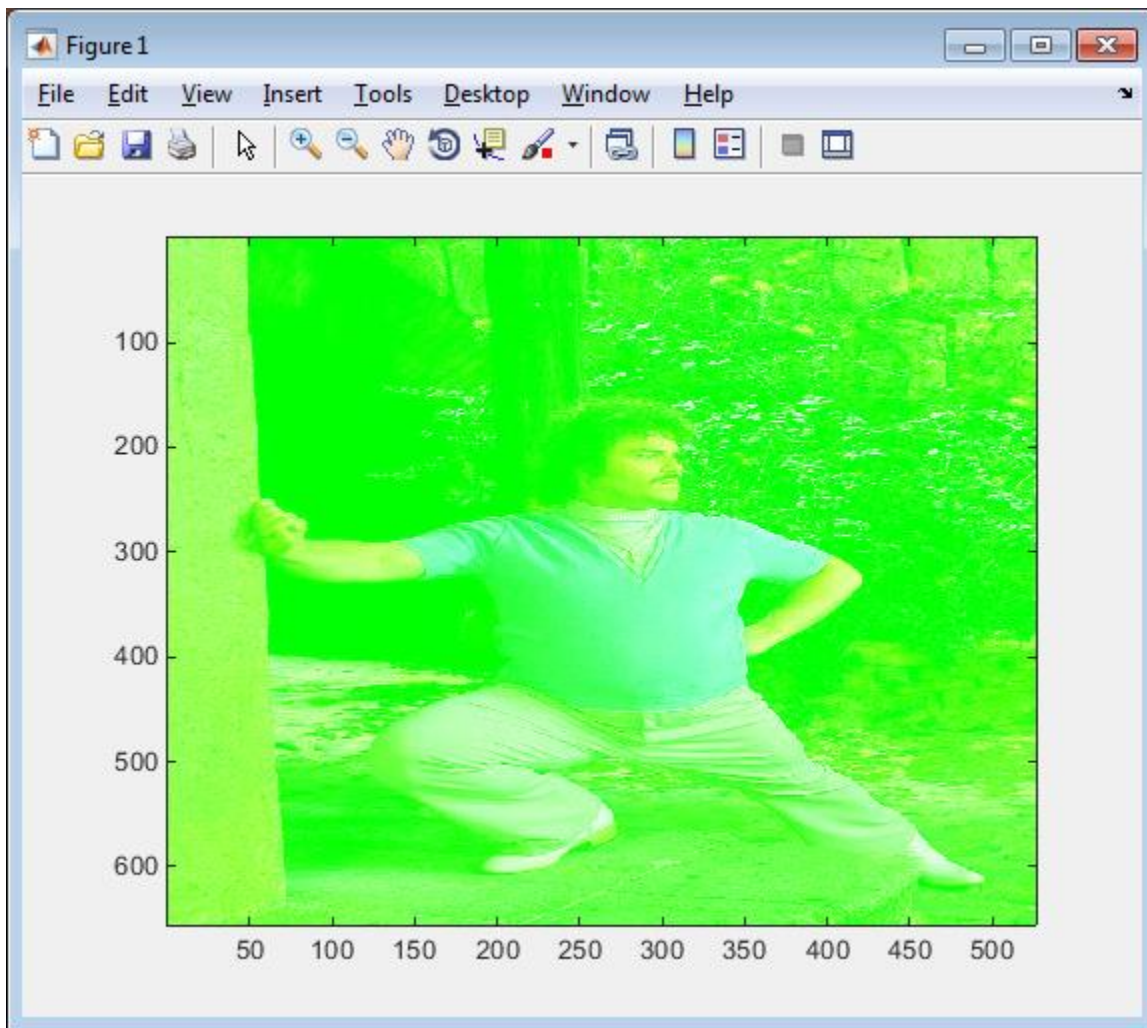
**Test Cases:**

```
>> load Test.mat  
>> imagesc(y);
```

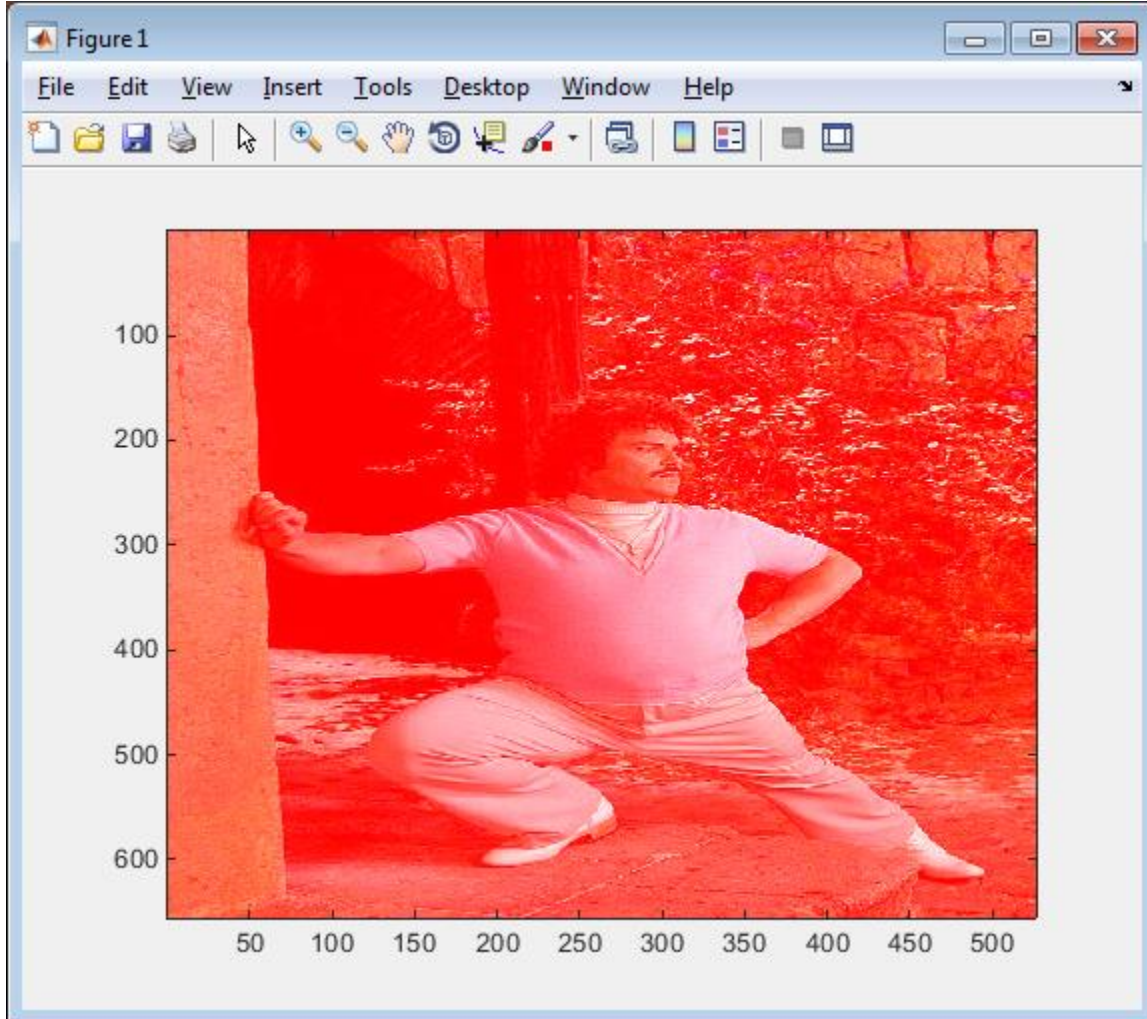




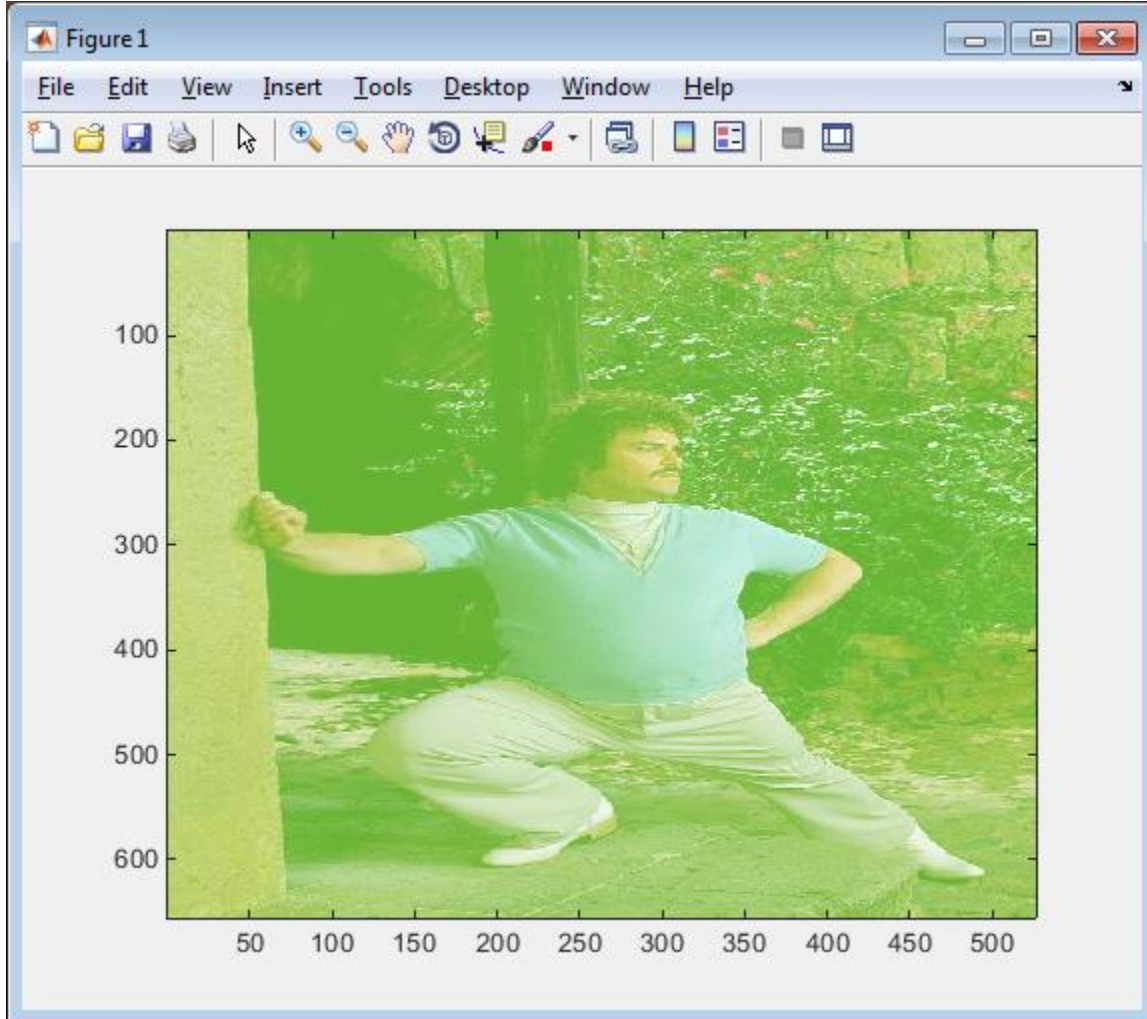
```
>> imagesc(myIntensity(y,[0,1,0]));
```



```
>>imagesc(myIntensity(y,[1,0,0]));
```



```
>>imagesc(myIntensity(y,[.4,.7,.2]));
```



**Deliverables:** Submit the following m-files (separately) onto Blackboard. **Be sure that the functions are named *exactly* as specified, including spelling and case.**

myRotate90.m  
myInvertColors.m  
myIntensity.m