EE 569: Homework #3

Issued: 10/3/2014 Due: 11:59PM, 11/9/2014

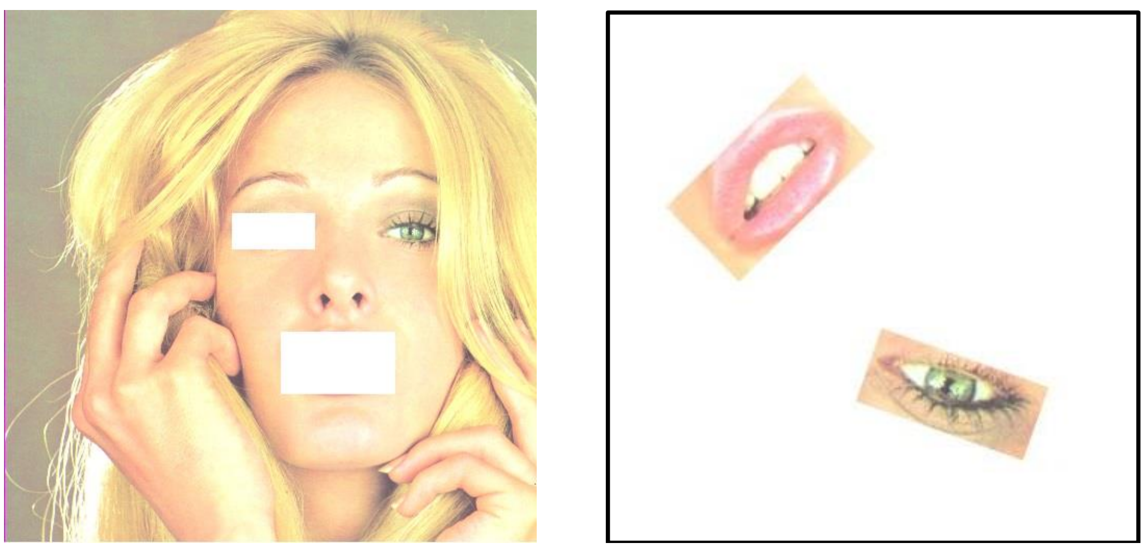
General Instructions:

1. Read Homework Guidelines and MATLAB Function Guidelines for the information about homework programming, write-up and submission. If you make any assumptions about a problem, please clearly state them in your report.
2. You need to understand the USC policy on academic integrity and penalties for cheating and plagiarism. These rules will be strictly enforced.

Problem 1: Spatial Warping (35 %)

(a) Puzzle Matching (Basic: 15%)

You are given two images ("Tiffany.raw" and "Components.raw") as shown in Figure 1. Apply geometric transformations to the two components and move them back to the two holes in the Tiffany image.



(a) Tiffany.raw (b) Components.raw

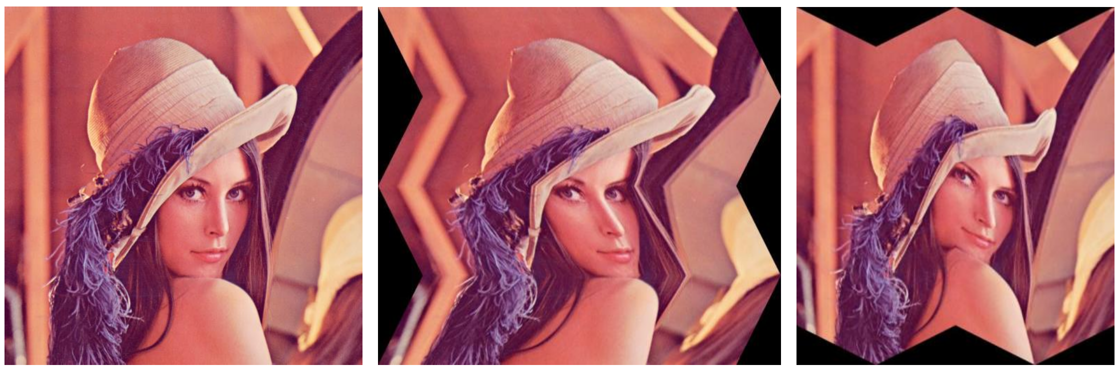
Figure 1: An image puzzle exercise.

You can solve this problem with the following steps:

1. Write a program to determine the center coordinates and the dimension of the holes in the Tiffany image, and report your results.
2. Write a program to determine the center coordinates, the orientation angle and the dimension of the components in the components image, and report your results.
3. Write a geometrical transformation program to manipulate the components in the following three steps in sequence: 1) rotation, 2) scaling and 3) shift. Show the intermediate results of the components image after each step.
4. Write a program to overlay the two images to recover the original Tiffany image, and show the final result.

(b) Shear Effect (Basic: 10%)

You can create some special visual effect using the geometric transformation. One example is shown in Fig. 2, where the input Lina image is under the horizontal shear and the vertical shear. The sizes of the input Lena, the horizontal shear Lena and the vertical shear Lena are 512x512, 576x512 and 512x576, respectively. Write a problem to implement the horizontal shear and the vertical shear special effects and apply them to the Jennifer image in Fig. 3. Describe your method and present the result in your report.



(a) Original Lena (b) Horizontal shear Lena (c) Vertical shear Lena

Figure 2: Three Lena images: (a) original (b) with horizontal shear and (c) with vertical shear.

Figure 3: jennifer.raw

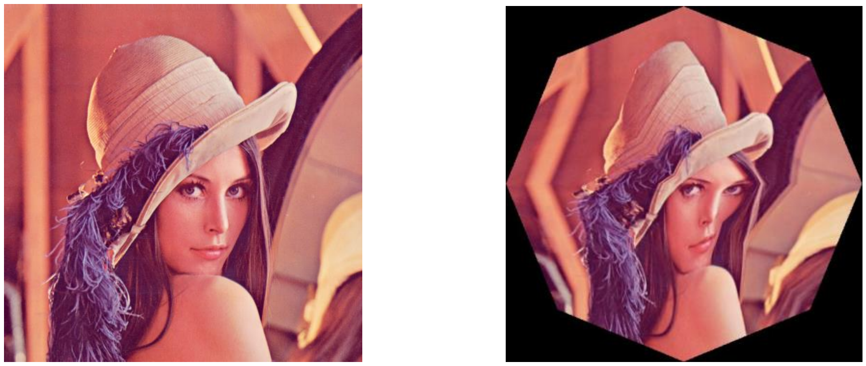
(c) Square-to-Octagon Warping (Advanced: 10%)

Design a warping algorithm that maps a square image to an octagon-shaped image. An example is shown in Figure 4. Note that the output image should satisfy following requirements.

1. Pixels that lie on boundaries of the square should still lie on the boundaries of the octagon.
2. The center of original images should be mapped to the center of warped images.

Describe your approach, apply the algorithm to the fruit image in Figure 5, and show the resulting images.

Next, develop a reverse warping algorithm to the warped fruit image to recover the input one. Compare the recovered and the original images. Is there any difference between them? If any, explain the source of distortion.



(a) Input Lena (b) Warped Lena

Figure 4: Warp the original Lena image to an octagon-shaped image.

Figure 5: fruits.raw

Problem 2: Optical Character Recognition (35 %)

(a) Learning from Ideal Reference Symbols (Basic: 15%)

A reference symbol set is given in Figure 6. You are asked to develop a decision tree classifier to partition the symbol set (the root of the tree) to reach each individual symbol (the leaf of the tree).

1. Segment each character into a rectangular region of interest (ROI).
2. Binarize each ROI into a black/while image.
3. Extract features from each binary image obtained in Step 2. You can use properties taught in class or described in the textbook, such as [1] Area, [2] Perimeter, [3] Euler Number, [4] Circularity, [5] Spatial Moment, [6] Symmetry, and [7] Aspect Ratio.
4. Build a decision tree for each segmented symbol using extracted features in Step 4.

S E L B Y P D C
e l y 1 2 3 4 5 6 7
8 9 0 / :

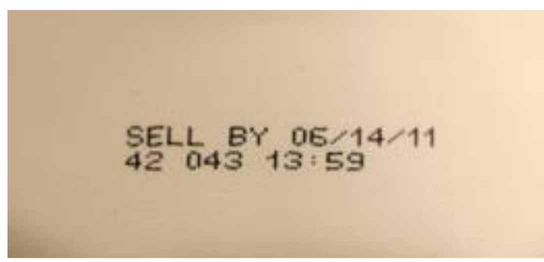
Figure 6: A reference symbol set.

Please report the extracted features and the decision tree used in your OCR program.

(b) Test on Real World Images (Basic: 10%)
Two test images are given in Figures 7(a) and (b). Use your OCR program built from part (a) to identify each symbol in the test images. For test image (a), you only need to take care of the date information (including "SELL BY:") inside the red circle and above the long horizontal line.

Hint: You need to refine your segmentation algorithm to separate each symbol well in these non-ideal images. Please describe your approach and evaluate the performance of your OCR program in your report.

(a) OCRtest1.raw



(b) OCRtest2.raw

Figure 7: Two OCR test images.

(c) Test on Symbols on Curved Surfaces (Advanced: 10%)
Symbols may become distorted when they are printed on curved surfaces. Two examples are shown in Figure 8. Your program should be able to recognize the date printed on the bottles, including the month abbreviation "SEP" and "DEC automatically. Build the association with each test symbol and the reference symbol.
To facilitate the segmentation task, you need a better preprocessing technique for image quality enhancement. Describe the developed preprocessing algorithm in your report.

(a) OCRmilk1.raw (b) OCRmilk2.raw

Figure 8: Recognition of **s**ymbols on curved surfaces.

Problem 3: Texture Analysis and Segmentation (30%+10%)

In this problem, you will conduct texture analysis and segmentation two tasks.

(a) Texture Classification (Basic: 10%)

Construct twenty-five 5x5 Laws Filter using the filters in Table 1.

Table 1: 1D Kernel for 5x5 Laws Filters

Name	Kernel
L5 (Level)	[1 4 6 4 1]
E5 (Edge)	[-1 -2 0 2 1]
S5 (Spot)	[-1 0 2 0 -1]
W5(Wave)	[-1 2 0 -2 1]
R5 (Ripple)	[1 -4 6 -4 1]

Twelve sample images, sample1.raw to sample12.raw (size 256x256) are provided for the texture classification task. Samples of these images are shown in Figure 9. Categorize them into four texture types, and each type contains three images. You should follow the following steps:

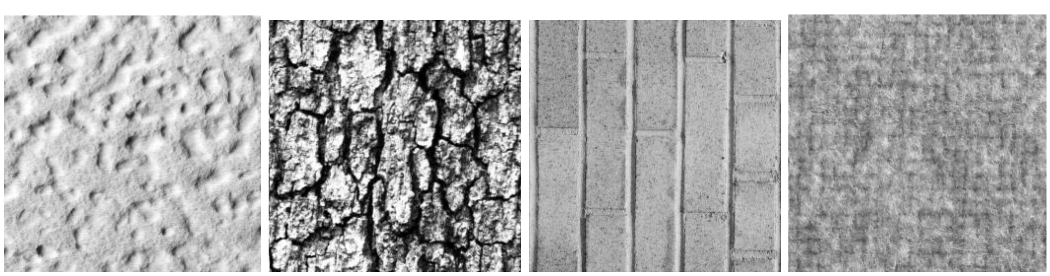


Figure 9: Class Representatives of sample[N].raw

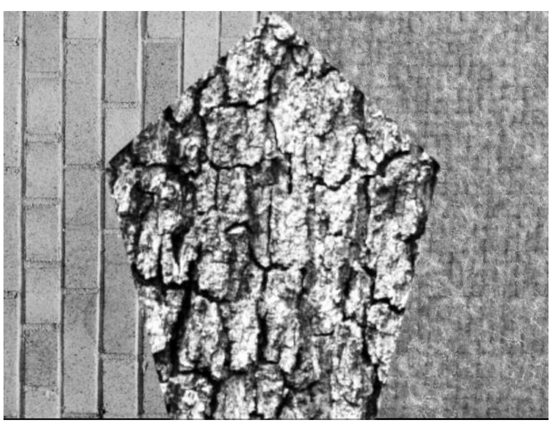
1. Use twenty-five 5x5 Laws Filters to extract feature vectors from each pixel in the image (use appropriate boundary extensions).
2. Average the feature vectors of all image pixels, leading to a 25-D feature vector for each image. Which feature dimension has the strongest discriminant power? Which has the weakest? Please justify your answer.
3. Reduce the feature dimension from 25 to 3 using the principal component analysis (PCA). Plot the reduced 3-D feature vector in the feature space.
4. Use the K-means algorithm for image clustering based on the 25-D and 3-D obtained in Steps 2 and 3, respectively.

You may use built-in C/Matlab functions of PCA. Report and analyze your results.Report and analyze your results.

(b) Texture Segmentation (Basic: 20%)

Apply twenty-five 5x5 Laws Filters to do texture segmentation for the image shown in Figure 10 using the following steps.

1. Laws feature extraction: Apply all 25 Laws filters to the input image and get 25 gray–scale images.
2. Energy feature computation: Use a window approach to computer the energy measure for each pixel based on the results from Step 1. You should try different window sizes. After this step, you will obtain 25-D energy feature vector for each pixel.
3. Energy feature normalization: Except for $L5^T L5$, all other kernels have a zero-mean. $L5^T L5$ is not a useful feature for texture segmentation. Use its energy to normal all other features at each pixel.
4. Feature dimension reduction: Apply the PCA algorithm to reduce the feature vector dimension from 25 to 9.
5. Segmentation: Use the k–means algorithm to perform segmentation on the composite texture images given in Figure 10 based on the 9-D energy feature vectors.


(a) com_1.raw

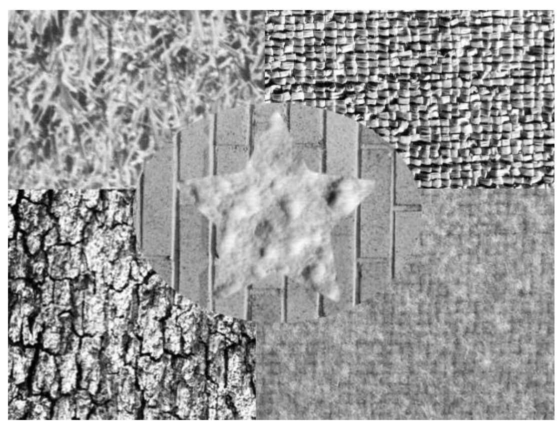

(b)com_2.raw

Figure 10: Two texture mosaic images: (a) simple and (b) complicated images.

If there are K textures in the image, your output image will be of K gray levels, with each level represents one type of texture. For example, there are six textures in Figure 10(b) you can use six gray levels (0, 51, 102, 153, 204, 255) to denote six segmented regions in the output image.

(c) Advanced Texture Segmentation Techniques (Bonus: 10%)

You may not get good segmentation results for the complicated texture mosaic image in Figure 10(b). Please develop various techniques to enhance your segmentation result. Several ideas are sketched below.

1. Consider the segmentation task in two spirals – the first spiral gives the coarse segmentation result while the second spiral offers fine-tuned segmentation results.
2. Develop a post-processing technique to merge small holes.
3. Enhance the boundary of two adjacent regions by focusing on the texture properties in these two regions only.
4. We reduced the feature vector dimension from 25 to 3 and 9 in (b) and (c), respectively. This choice is however heuristic. Discuss a way to choose the PCA reduced feature dimension.

Appendix:

Problem 1: Spatial Warping			
Tiffany.raw	512x512	24-bit	color(RGB)
Components.raw	350x350	24-bit	color(RGB)
fruits.raw	512x512	24-bit	color(RGB)
jennifer.raw	512x512	24-bit	color(RGB)
Problem 2: Optical Character Recognition			
OCR_training.raw	600x300	8-bit	gray
OCR_test1.raw	282x181	24-bit	color(RGB)
OCR_test2.raw	265x120	24-bit	color(RGB)
OCR_milk1.raw	230x120	24-bit	color(RGB)
OCR_milk2.raw	231x300	24-bit	color(RGB)
Problem 3: Texture Analysis, Segmentation and Synthesis			
sample1–12.raw	256x256	8-bit	gray
com_1.raw	600x450	8-bit	gray
com_2.raw	600x450	8-bit	gray

Sample Codes Provided

kmeans.c	C code provided for k-means algorithm

References

[1] Alexei A. Efros. *Image Quilting for Texture Synthesis and Transfer.* http://graphics.cs.cmu.edu/people/efros/research/quilting.html

Reference Images

All images in this homework are from Google images or the USC-SIPI image database.