

## CS303B Assignment-1

### Aim

The book cover in Figure 1 is a composite image of the Mona Lisa; it consists of many tiles each of which is a painting. The aim of this assignment is to develop, evaluate and report on image processing software to automatically create composite images of various types, using images automatically selected as being of certain classes of scene.

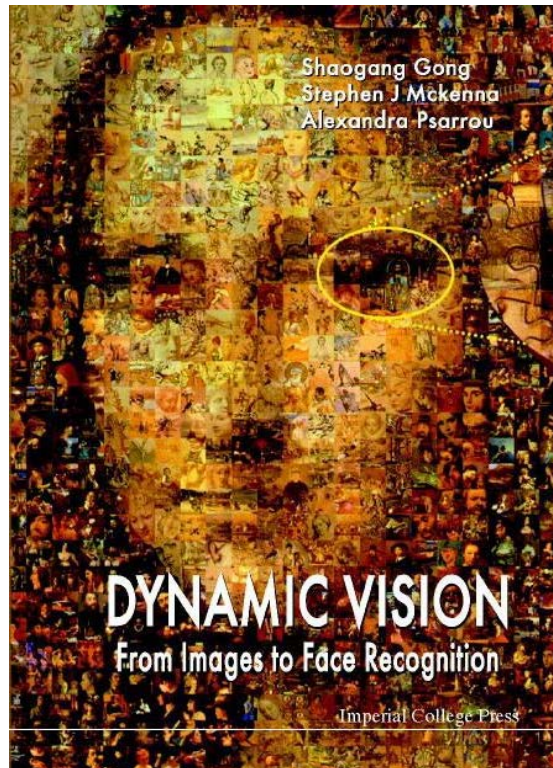


Figure 1

### Terminology

The image to be approximated is called the *target image*. The images used to create the tiles are called *source images*. In Figure 1, for example, the Mona Lisa is the target image, and the source images are various paintings.

### Image Data

You will be provided with sets of images of several hundred outdoor scenes. Each image has been labelled as either *natural* or *manmade*. The *natural* images are of scenes dominated by mountains, hills, desert, or sky, for example. The *manmade* images are of scenes dominated by commercial buildings, shops, markets, cities, and towns. You should use these image sets for each Part of this assignment. The images sets are divided into disjoint subsets for *training* and *testing*.

Training set:	500 <i>natural</i> images & 500 <i>manmade</i> images
Test set:	250 <i>natural</i> images & 250 <i>manmade</i> images

## PartA 基本要求

1. num of pixels
2. num of tiles
3. can use other image sources

**PART A: Composite image generation**

The first part of your assignment is to develop software capable of creating composite images.

The user should be able to select the number of pixels and the number of tiles in the output composite image. You should use the training set of 1000 images to obtain source images.

*Optionally, you might want to source other image sets to create your own composite images. If you do so, make sure that their use is not prevented by regulations (e.g., specific stipulations on some websites limiting the use of published material) or decency considerations.*

Source images should be cropped and have their sizes adjusted appropriately prior to inserting them into the composite image (e.g. by Gaussian smoothing and resampling).

Which source image to use for a given tile should be determined by computing a *measure of similarity* between the tile in the target image and each source image. You should design and implement an appropriate similarity measure. A good starting point is to compute image histograms and use a measure of histogram similarity. However, obtaining satisfying result might require taking into account other characteristics of the images such as their texture and structure. You might even consider taking into account neighboring tiles when deciding which source image to use.

考虑的方法

- 1, source images 可以被裁剪
- 2, 通过比较相似度 (需要自己设计以及实现)
  - 通过histogram
  - consider their texture/structure
  - 考虑周边的tiles 从而进行选取

**PART B: Binary image classification**

PartB 基本要求:

- 1, classification of natural and manmade
- 2, 实现knn
- 3, 用partA 中的similarity 开始, 设计一个不同的分离器
- 4, 通过评估performance
- 5, 不可以用测试集

The second part of your assignment is to develop and quantitatively evaluate software to classify images as belonging to one of two classes: *natural* or *manmade*. You should implement a K-nearest neighbor classifier based on an image-image similarity measure.

Start by using the similarity measure you developed in PART A. Subsequently, you might want to design a different similarity measure better suited to this classification task. (Think about what differentiates these two classes of image and about methods you have learned about in this course.)

You should evaluate the performance of your classification system on the test set. Test results should be given in terms of **computation time, misclassification rates for each class, and overall accuracy**. Of course, the system must not make use of images from the test set when making classification decisions.

PartC

- 1, 通过combine前两个part 得到不同class产生的图片
- 2, target image -> classify -> build a image only from the same class

**PART C: Class-specific composite images**

In this part, you should combine what you have done in Parts A and B to produce class-specific composite images. Given a target image, your system should first classify it, and then build a composite of it using only images from the same class (as determined using your classifier from Part B).

**PART D: Competition**

During week 9 we will run a competition in which each team's image classifier will be directly compared in a controlled setting. You will be given a new set of 250 *natural* and 250 *manmade*

images on which to test your system. You can use all 1500 previously available images as training images. You should submit the test classification results along with the code you used to produce them.

## ***Teamwork and Assessment***

This assignment will be assessed based on written reports, presentations, and competition results. This will include an element of moderated self/peer assessment.

The class will split into **teams**. Each team should produce its own **integrated** solution. Every team member should contribute to some aspect of all three of the following: (i) software development (ii) report preparation (iii) presentation/competition.

### **Competition** (*Thursday of week 9*)

This will be held during the **lab session**. The results will be collated and presented to the class by the lecturers during week 12.

### **Team Report** (*due Thursday of week 9*)

Each team should **submit a technical report (approximately 15-20 pages including images)** both electronically and as a **hardcopy**. The report should describe the algorithm design, implementation, testing and evaluation of the software developed. Clear justification should be given for the choice of methods compared with possible alternatives. References to the literature should be made to set your project in context and cite relevant work. You should use an accepted citation style. Code produced should be submitted in electronic form. Your team might want to appoint an editor for the report but every member of the team should contribute material to the report. You should submit the electronic version of the team report and code to VLE.

Team Report 要求  
1, 电子版+纸质版  
2, 15-20面  
3, 包括(算法设计, 具体实现, 测试评估), 不同方法之间的比较  
4, 文献引用, APA  
5, code需要被上交  
6, 每个人的都要写

### **Individual Report** (*due Thursday of week 9*)

Each student should also submit a short individual report. Half a page is all that is required; it should certainly be no longer than a page. This individual report should be submitted to VLE. It need not be seen by any other students. It should contain:

1. An assessment of the extent to which you contributed to the outcome of your team, including the three ways in which you personally contributed most to the project.
2. A brief assessment of the contribution of each member of your team.
3. A suggested overall mark (%) for this assignment for each member of your team, including yourself.

Individual Report  
1, half page, 不长于一面  
2, 贡献比, 三种对整个project 贡献最大的部分  
3, 对于队友进行评估  
4, 整体分数

pre 要求

- 1, 为什么选择这个方法
- 2, 最终系统的结果, 以及对未来工作的建议
- 3, 每组15min
- 4, 以每个人的介绍以及各自contribution 开始

### **Presentations** (*Thursday Week 9*)

Each team will give a short presentation in which choices of method should be carefully justified in the context of alternative methods that could have been used. The final system design and results should then be presented before drawing conclusions and making suggestions for future work. Each team member should contribute (either in preparation or delivery or both). Each team will be allocated around 15 minutes plus 5 minutes for questions and discussion. The presentation should begin by introducing the team members and stating briefly their contributions.

**Marking Scheme**

This assignment counts for 30% of the total module mark

不懂这个评分是怎麼樣的

For the purposes of marking, the relative weighting of Parts A, B and C is 45:45:10

The relative weighting of the team report, the competition results, and the presentation is 70:10:20.

The individual reports will be used to assist the markers in the relative marking of team members, in combination with the markers' own observations of performance.

*Jianguo Zhang*  
Sept 2020