Readings/ References

William K. Pratt, Introduction to Digital Image Processing," CRC Press, 2013

(available online at: https://ie.u-ryukyu.ac.jp/~asharif/pukiwiki/index.php?plugin=attach&refer=%BF%AE%B9%E6%BD%E8%CD%FD%CF%C0&openfile=Digital%20Image%20Book2_Prat.pdf)

Recommended chapters: 1,2, 7, 8, 10, 13, 14

Preparation:

- Bring a laptop
- Familiarise yourself with Open CV library (more details below). Choose one programming language that you are familiar/ comfortable with and download the libraries. Maltlab and Python are usually preferable than C++, if you don't have any previous programming experience.
- Download the subset of Places dataset (needed for the exercise) (~45MB) https://www.dropbox.com/sh/riw3s6nc8z7l6pw/AACnCnOEM1XGkMQKyDoRGTz1a?dl=0

OpenCV

Exercise will be based on OpenCV (Open Source Computer Vision Library: http://opencv.org. Open CV is an open-source BSD-licensed library that includes some hundreds of computer vision algorithms. OpenCV has C++, C, Python and Java interfaces and supports Windows, Linux, Mac OS, iOS and Android.

Information on installation and setup of OpenCV for a variety of platforms is available at: http://docs.opencv.org/2.4/doc/tutorials/introduction/table_of_content_introduction/table_of_content_introduction.html

Information on the core functionality of OpenCV is available at: http://docs.opencv.org/2.4/doc/tutorials/core/table_of_content_core/table_of_content_core.html#table-of-content-core

Hands-on exercise

1. Obtain data:

Download the provided sub-set of "Places" dataset described by Zhou et al. This includes a training set, testing set and class labels.

Dataset Reference: B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva. Learning Deep Features for Scene Recognition using Places Database. Advances in Neural Information Processing Systems 27 (NIPS), 2014.

2. Feature extraction (Part 1):

Choose a set of reasonable features to extract from the images and visualise the effect. These can include:

- Appearance features
- Image filters
- Local features
- Colour features

3. Classification (Part 2):

Using the training set, train a linear SVM using your features and calculate classification accuracy on testing set both using single and multimodal features. Compare different models performances.

LibSVM is an open source library for SVM implementation: http://www.csie.ntu.edu.tw/~cjlin/libsvm/
There is also an openCV tutorial for
SVMs: http://docs.opencv.org/2.4/doc/tutorials/ml/introduction_to_svm/introduction_to_svm.html

If you have time, try simple Neural Network architecture for the classification task in hand.