ME 592X: Data Analytics and Machine Learning for Cyber-Physical Systems

Homework 2

Homework Assigned on January 23, 2018 Homework Due on: February 1, 2018
One submission per group

Motivation

This homework is to provide an experience of Data Preparation involved in Data Analytics for Cyber-Physical Systems.

General Instructions

The dataset and problems for each group are slightly different, but the motivation remains the same. Following are some instructions for all the theme groups. Specific instructions for each group shall be provided in the relevant sections.

1. The final code must be pushed to git before the deadline and relevant data for running the codes should be placed at `/ptmp/ME592_2018_GRPNAME` in hpc-class cluster. The reason for this is to make the git repository compact and generic for any data.
2. Use the discussion board in Canvas in case of any issue.

Expected Outcome

1. A code pushed in git to preprocess each dataset provided.
2. A report/presentation (8 minutes presentation + 2 minutes questions) of your problem and results; You will present this report as discussed in class.

.

Self-Driving Cars

The following problems are based on a particular dataset very popular in the Self-driving cars community - Oxford RobotCar Dataset. Over a period of one year, an autonomous Nissan LEAF traversed through a 10km route in central Oxford twice a week for recording driving with over 20 million images collected from 6 cameras mounted to the vehicle along with LIDAR, GPS and INS. The data was collected in all weather conditions, road and building works over the perious of an year. This data is useful for long term localization and mapping. Now, we shall see some of the techniques used for processing the data. Download the large sample data(1GB size) from `http://robotcar-dataset.robots.ox.ac.uk/downloads/`. Using the software development kit provided in the same webpage, perform the following tasks:

1. Image Demosaicing and Undistortion of the Images

2. Using the LIDAR, GPS/INS and Visual Odometry generate the 3D Point Cloud

3. Project the generated point cloud on the camera image

You can use the documentation at `http://robotcar-dataset.robots.ox.ac.uk/documentation/` for reference on how to use the software development kit.
Finally, a video of all the final images stitched together should be a good deliverable.

Energy and Power Analytics

The .mat files in the electricity dataset contains energy consumption (in Watts) time series for different end uses as well as the main power for a house sampled at 1Hz.

The task is to explore the efficacies of different distance metrics and transformations for time series by computing differences among different end uses and the main power. Perform the following:

1. Perform pre-processing of all the variables (such as normalization and denoising).
2. Use direct Euclidean distance metric.
3. Convert the data to frequency domain using fft and then do a comparison using euclidean distance metric.
4. Use KL Divegence metric and perform comparison.
5. Compare the data in wavelet transformed space.
6. Use windowed spectrogram to identify motifs in the main power data to detect changes in time-series charecteristics.

Comment on your findings of the comparison and about the feature detection.

Design and Manufacturing

Finite Element Analysis is a significant part of the design process. Often running simulations repeatedly for slight changes in geometry is expensive, specifically when we know that it is just going to be another iteration of the design process. The dataset in DM.zip contains the following:

1. 64 input geometries made of NURBS surface. Each input geometry contains 3 smesh files for three surfaces which are interacting in the analysis.

2. Deformed geometry(at 80th time step and 140th time step of the analysis) for the 64 geometries at 5 temperature and 3 pressure conditions.

Each of the geometry is placed in folder run1-64. Each folder contains three smesh files. Each of the smesh file contains the following:

- First 4 lines talk about the no. of components, the DOF, number of nodal points and something about the post-processing. You might find it easy to skip these 4 lines while parsing through the file.

- After the line 4, x,y and z coordinates of each node is specified. For simplicity, all the geometries are having 17 and 12 nodes in two directions of the surface.

Each of the final geometry is named as result_tstep_temp_pressure_geometry, where tstep refers to the time step of the deformed geometry (80 and 140). temp refers to the 5 temperatures (300K, 350K, 400K, 450K, 500K) and pressure refers to 3 pressure conditions (76mmHg, 80mmHg, 84mmHg) and geometry refers to the 64 geometries. Each file contains the final deformation of all the nodes of the three surfaces in the same order as the smesh files (skip line 1 which contains time step and analysis relevant details).

Using the following data, perform the following tasks:

1. Create two ordered pairs of (input,output) corresponding to two time steps. Here, input refers to tuple of (geometry, temperature, pressure). Since, each of the geometry contains 17×12 nodes. Construct an array of shape $17 \times 12 \times 3$ for each geometry. Shape of one element of the ordered pair may look something like ([[17,12,3],[1,],[1,]], [[17,12,3]]).

2. Consider the input geometry and output geometry to be images with three channels (x, y and z channels instead of RGB channels). Since, the range of x, y and z channel are completely different, we would like to apply PCA whitening to normalize the data. Refer to `http://ufldl.stanford.edu/wiki/index.php/Implementing_PCA/Whitening` for more information on how to whiten an image. You would need to whiten all the input geometries, output geometries of both the time steps.

3. Also, do ZCA whitening instead of PCA whitening.

4. Once, the output images are whitened, vectorize the output images of time-step 80 and use that for plotting t-SNE. While plotting the t-SNE, you would need to create three plots to mark the labels of the data based on geometry, temperatures and pressures. Comment on the data distribution based on the t-SNE results in terms of geometry, temperatures and pressures. You could use any of the implementations available at `https://lvdmaaten.github.io/tsne/`.

5. Repeat the same process on time-step 140 as well. Comment if the inference on data distribution changes for this case.
6. Also comment on differences in observations made from using PCA whitening and ZCA whitening.

Engineering Imaging Analysis

1. Image enhancement Camera sensors are often limited in ability to clearly capture objects without defects from images or videos taken in a poorly-lit environment. The goal in many applications is to enhance the brightness, contrast and reduce noise content of such images in an on-board real-time manner. Popular methods for enhancing low-light images used are Histogram Equalization, Contrast-limiting adaptive histogram equalization (CLAHE), Gamma adjustment, 3D block matching (BM3D). All the tools are available online for usage.

There are 16 image pairs (original image, corrupted-dark image) in image.zip. Use above mentioned four methods for enhancing the corrupted-dark images and compare using the following metrics.

1. Peak signal-to-noise ratio (PSNR)
2. Structural similarity index (SSIM)
3. Earth mover's distance (EMD)

Now, flatten all the images (16 original, 16 corrupted-dark image, and 4×16 processed images) and plot the t-SNE for this 96 data points. Comment on the several statistics on each of the metric and the t-SNE plot on the given 16 images.

2. Image Segmentation The goal here is to extract only the Arabidopsis seedlings from the images for more analysis. An example is shown in Fig. 1. Write a script to perform the same operation on the given two images to obtain a clean version of the images.

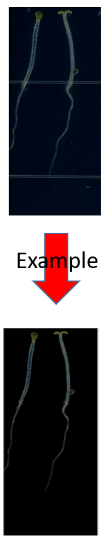

Figure 1: Example of image transformation required for problem 2

Robotics

In robotics, often we encounter dataset containing RGB-D data and SLAM and Visual odometry systems are necessary for making decisions. For this assignment we would use RGB-D SLAM Dataset and benchmark available at `https://vision.in.tum.de/data/datasets/rgbd-dataset`. The dataset contains the color and depth images of a Microsoft Kinect sensor along the ground-truth trajectory of the sensor. The data was recorded at full frame rate (30 Hz) and sensor resolution (640×480). The ground-truth trajectory was obtained from a high-accuracy motion-capture system with eight high-speed tracking cameras (100 Hz). Among the many sequences of RGB-D data for several tasks, please use the sequence of large_cabinet in the category of 3D Object Reconstruction. The tasks to perform are as follows:

1. The color and depth images are provided in an un-synchronized way. We would need to associate both of them and find best matches based on the time stamps.
2. Generate point cloud from the RGB and Depth information of associated images of previous step.
3. Add the point clouds to ROS bag files and visualize the dataset in RVIZ

You may use the codes and documentation provided at `https://vision.in.tum.de/data/datasets/rgbd-dataset/tools` for completing this task.

Attachments

1. EPA.zip
2. DM.zip
3. image.zip