CMPT 318 Project - Webcam, Predictions, and Weather

1- Objective

Given a database of webcam images and a database of weather information, the machine will be trained for the ability to predict weather with given image.

2- Required library

- I/O library used: sys, glob, shutil, os
- Open-cv
- Numpy
- Pandas
- Sklearn
- Matplotlib

3- Data Source

Images
- Total of 6991 webcam images collected are taken at English Bay in downtown Vancouver by Kat Kam (http://www.katkam.ca/) and scaled to 256×192. Images collected are taken once every hour from June 5, 2016 to October 31, 2017.

Weather data
- Weather data from May, 2016 to October 2017 is collected from Canadian Government (http://climate.weather.gc.ca/historical_data/search_historic_data_e.html) in csv file format

4- Data Cleaning Process

Data cleaning based on information availability
- Combine all weather data csv files into one dataframe
- Extract necessary data column such as 'Year', 'Month', 'Day', 'Time', and 'Weather'
- Remove rows with no weather description
- Generate image file names based on the value of year, month, day, and time
- Get images and store into the corresponding rows in the dataframe
- Remove rows with no image available

Data cleaning based on data quality
- Calculate the mean of all pixel values to get the darkness of the image
- Remove rows with image darkness < 65 [based on experiment] as these images are too dark to predict weather

- Remove rows with weather description which is 'Thunderstorms', 'Freezing Fog', 'Hail', 'Snow Pellets' as these data only have around 2-3 images available
- Extract images with "more than 10,000 blue pixels", "ratio between blue and white less than 0.5", and "difference between red, blue, and green values larger than 29" [based on experiment] found out these images are all clear sky, but some descriptions says cloudy or rain, therefore, descriptions change to clear
- Finally, manually check if description and image fit
- Around 2700 images after data cleaning

Features extraction
- darkness (mean of all pixel values)
- Red, Green, Blue (mean of pixel values in each channel)
- White pixel count (total number of white pixels [larger than 200 in grayscale])
- Blue pixel count (total number of blue pixels [in range of (95, 120, 180) to (180, 210, 255)])
- Ratio between white and blue pixels (white pixel count : blue pixel count)
- RGB difference (distance between red, green, blue value for the purpose of distinguishing color pixels and grayscale pixels)

5- Training Model

- Support Vector Classifier (SVC) with linear kernel is used for machine learning model

6- Result
- The maximum accuracy achieved is 76.3% when testing on 270 random images
- Some images are hard to distinguish between rain and cloudy even with human eyes, so the prediction may be correct whereas the actual weather description may be wrong.

7- Visualizations

8- Limitations
- The weather description collected around YVR airport may not match the images collected at English Bay, that could lead to false information. (Figure 1)
- All descriptions are converted into 4 categories such as "rain", "clear", "cloudy", and "snow"
- Dark images at night are removed

9- Accomplishment Statement
- Used machine learning concept to implement weather prediction software
- Processed and cleaned data to extract useful information
- Accomplished the project within a short period of time

```
Actual weather: Clear
katkam-scaled/katkam-20160917160000.jpg
```
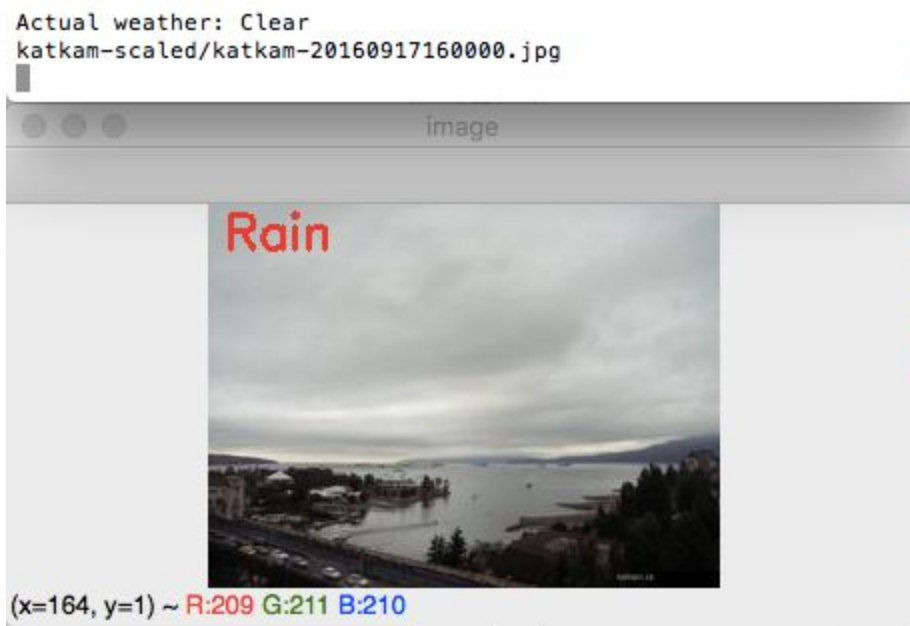
Rain

(x=164, y=1) ~ R:209 G:211 B:210

Figure 1. The weather description of this image is clear as the prediction is rain, the prediction seems more accurate than weather description in this case.