David Zikel - 766 Project
Midterm Report

1. Project review

This project focuses on determining ambient weather information from an image - mostly categorical guesses as to the season, time of day, etc. It is to determine this information from features of the image such as color balance and edge visibility. (For instance, redder colors indicate warmer weather, more lighting indicates daytime, clarity of edges corresponds with a lack of fog, etc.) All of this information will be determined with no human input besides the original creation of the program and the selection of the image.

2. Specification revisions

As suggested in the comments to the original project proposal, I have decided to scale down the scope of this project. Categorization of objects based on the arrangement of their edges is no longer being implemented, along with the creation of a (hardcoded or AI-driven) object classifier based on this information. Detection of snow given noisy object surfaces is also no longer part of the project. Using albedo to correct for shadows while finding object colors is still planned, however, as are the rest of the features with implementation details discussed in the current and future work sections of this progress report.

3. Current progress and immediate work

Little progress has been made in the way of code - however, the material and theory to be used has become more clear. Both edge detection and albedo extraction have been covered in class, with edge detection in particular already being showcased in homework and MATLAB code. From this, determining the sharpness of an edge can be done by computing the direction of the edge, taking the average color along lines parallel to the edge and slightly offset from it (for computing sharpness for an entire edge - to compute sharpness at a point, colors can simply be taken on points offset from the line), and measuring how significant the color difference is. For computing lighting information, public albedo-approximation algorithms can be used, and the image can also possibly be divided into two sections to accommodate the skyline - for this, it may be simpler and more effective just to calculate one horizontal position for this skyline and not to use a cluster-finding algorithm such as *k*-means.

4. Future direction

For future work on the project, most or all of the algorithms discussed above can be implemented. An analysis can be done to see if the more complicated features for these algorithms need to be added - for instance, the performance of the simple horizontal skyline divider discussed above can be analyzed to see whether or not it would be useful to implement a cluster-finding algorithm. Work also needs to be done on the program's user interface - this is likely to be a very simple "select a file" dialog followed by a window displaying the properties found as plain text, but work on this UI has not been started yet.

5. Potential roadblocks

Albedo extraction is a difficult task and does not appear to be included in MATLAB as-is - it may be necessary to find a pre-existing library implementing a relatively simple algorithm such as Retinex. If no such library can be found, it would also be possible to design the program to work without albedo information by taking the average color value on any object and assuming that specular highlights and shadows will roughly cancel each other out. In addition, if a choice of clustering algorithm needs to be made (to find objects or horizon lines), tradeoffs will need to be made between the algorithm's accuracy and efficiency, with better object outlines resulting in much slower performance.