# COMP2550/COMP3130 ANU Warmup Project Report

**Christopher Claoué-Long (*u5183532@anu.edu.au*)**
**Jimmy Lin (*u5223173@anu.edu.au*)**

## 1 Introduction and Overview

The task at hand for this warmup project was to implement a set of useful features for a computer vision task, using the open source Darwin framework for machine learning and the MSRC dataset. The desired outcome was a feature vector that allowed the machine learning algorithms to classify images with over 50% accuracy.

## 2 Method

First of all, we consider the texture of one superpixel. One possible implementation is to respectively discretize the lightness of marginal RGB value (9 for each interval length) and assign each pixel to those intervals based on their RGB lightness. The strength of this method is the tremendous informativeness and it does result in a satisfying accuracy increase (directly to arround 0.43) . However, that discretization method occupies too many dimensions of feature vector (87 attributes) to the extent that the classifier was badly sensitive to other features. Alternatively,we attempt the mean and standard deviation of absolute marginal RGB value (only 6 attributes). Expectedly, it produces a considerable rise with occupying fewer dimension of feature vector. (It will be demonstrated latter why occupation of less dimensions is vital.)

First of all, we consider the texture of one superpixel. One possible implementation is to respectively discretize the lightness of marginal RGB value (9 for each interval length) and assign each pixel to those intervals based on their RGB lightness. The strength of this method is the tremendous informativeness and it does result in a satisfying accuracy increase (directly to arround 0.43) . However, that discretization method occupies too many dimensions of feature vector (87 attributes) to the extent that the classifier was badly sensitive to other features. Alternatively,we attempt the mean and standard deviation of absolute marginal RGB value (only 6 attributes). Expectedly, it produces a considerable rise with occupying fewer dimension of feature vector. (It will be demonstrated latter why occupation of less dimensions is vital.)

The second significant set of features is the mean and standard deviation of relative lightness of one pixel's marginal RGB value, that is, the lightness difference between the pairs of Red and Green, Green and Blue, Blue and Red.

On top of that, some other features that lead to comparatively slight accuracy advances are added into the our feature vector as well. For instance, the average smoothness counts the mean of RGB lightness difference of all pixels to their neighbour pixels. And location of one superpixel, more specifically, evalutates the mean and standard deviation of x,y coordinate of one pixel in its located image. The idea of adding this set of feature our final selection comes from the semantic perceptions that the higher stuff is more likely to be sky and that the lower stuff is more likely be ground or grass.

What we discussed before is all about the features of one superpixel itself. To further enhance the accuracy, we may need to, when forming the feature vector of one superpixel, take into consideration the brief information of neighboured superpixel. That is an intuitively meaningful approach because even human cannot recognize a small region without perceiving its surroundings. Yet, the experiments we have accomplished till now does not indicate the effectiveness of that approach. One possible reason is that the features we pick out for this approach provide little information regarding the surrounded environment of one superpixel. By the way, we apply some algorithm optimizations to reduce the time cost for training of classifier with neighbour superpixel detection, from about 6 hours to 25 minutes. One way to fulfill this progress is to replace linear list with hash set to minimize the time for membership evaluation.

## 3 Results

Accuracy tables with different features implemented.

## 4 Discussion

Interpretation of results, what went well, what went wrong (overfitting?), what could be done better.

The first few features we thought to implement were of course to measure the average red, green and blue values of a superpixel, since this would allow us to start differentiating between extremely different superpixels. Adding a standard deviation of the pixels' colours compared to the average colours over the superpixel would also help differentiate further between superpixels of radically different texture/contrast. We also tried out getting the difference and absolute difference between the red, green and blue average values across the superpixel to further bring out features dependent on the superpixel composition. **CONTINUE HERE.**

Intuition, thoughts and discussion on the CV features
RGB values and std deviation, centre (position), average grayscale, difference between colours in a superpixel, neighbours, texture/contrast etc.

## 5 Results

Pictures, maybe accuracy tables with different features implemented?

## 6 Discussion

numerical analysis of your algorithm - emailed Stephen for clarification on this.
What is the performance of your algorithm on the training set compared to the test set? Is this result expected?
Why is it important to evaluate pixelwise accuracy instead of accuracy on the superpixels?
What do you think is more important, the features or the machine learning classifier?
Interpretation of results, what went well, what went wrong (overfitting?), what could be done better.

## 7 References

S. Gould. DARWIN: A Framework for Machine Learning and Computer Vision Research and Development. In *Journal of Machine Learning Research (JMLR)*, 2012.
K. Park and S. Gould. On Learning Higher-Order Consistency Potentials for Multi-class Pixel Labeling. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2012.