

Capstone 2

Zoe Shim

Introduction

- Skin cancer is one of the most common cancer in US
- Early detection => 99% five-year survival, distant stage = > 23%
- ∴ Faster more accessible melanoma detection is needed

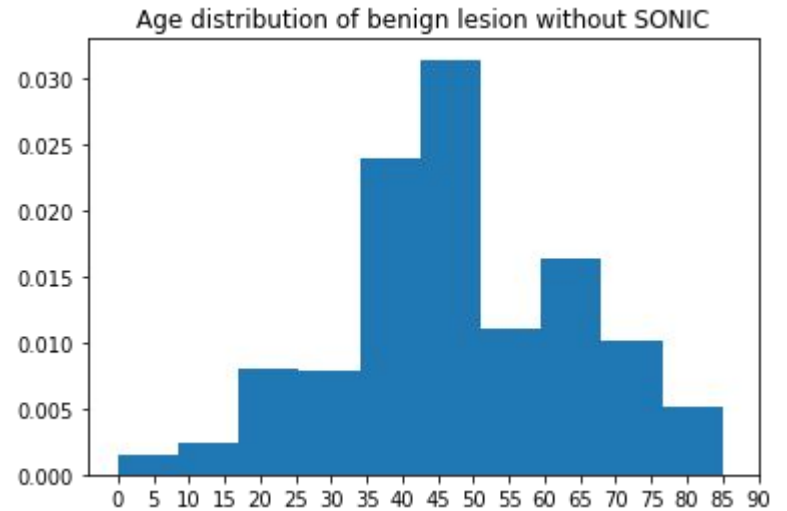
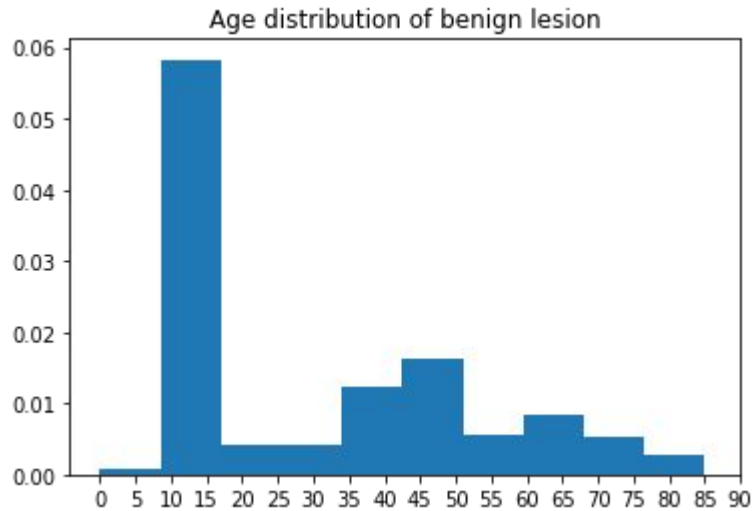
Data wrangling

- Download image data and their metadata from International Skin Imaging Collaboration
- Load metadata to dataframe form for EDA
- Load images and normalize
- Split the images' R,G,B channels for color analysis
- Apply gaussian filter and segment lesion using the threshold acquired via the filter
- Remove images that does not have enough contour to analyze

Data Exploratory Data Analysis

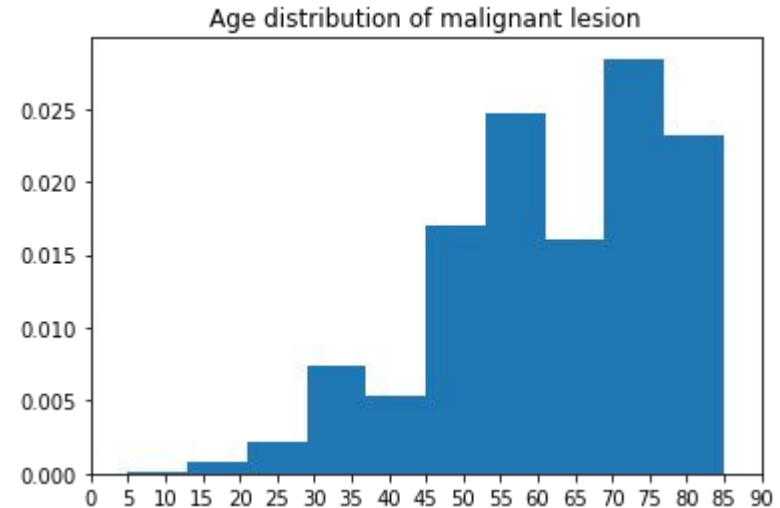
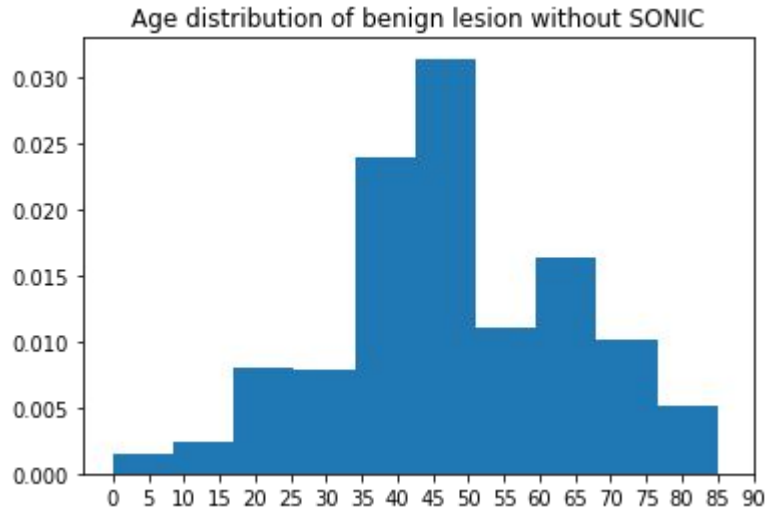
- Size of the lesion
 - Bigger the lesions, more likely it is to be malignant (D in ABCDE)
- Age of the patients
 - Removed certain dataset due to bias introduced
 - Older the patients, more likely to have malignant lesion
- Location of the lesion
 -

Benign vs. Malignant: Age distribution



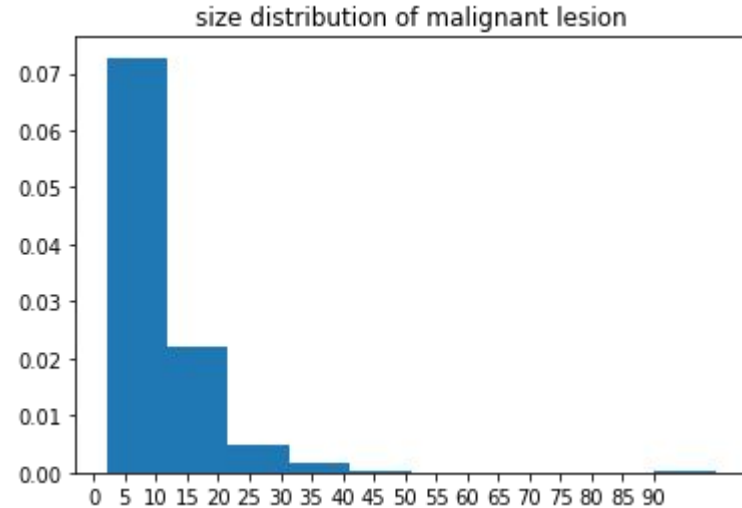
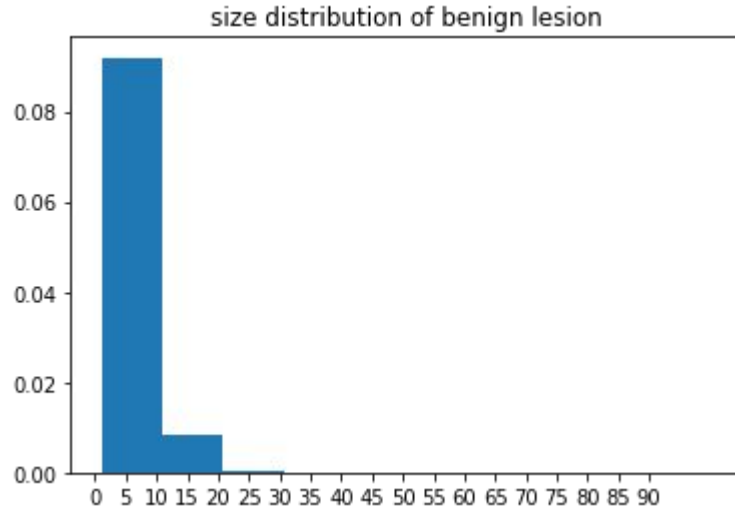
P value of independent T - test : 0

Benign vs. Malignant: Age - w/o some data



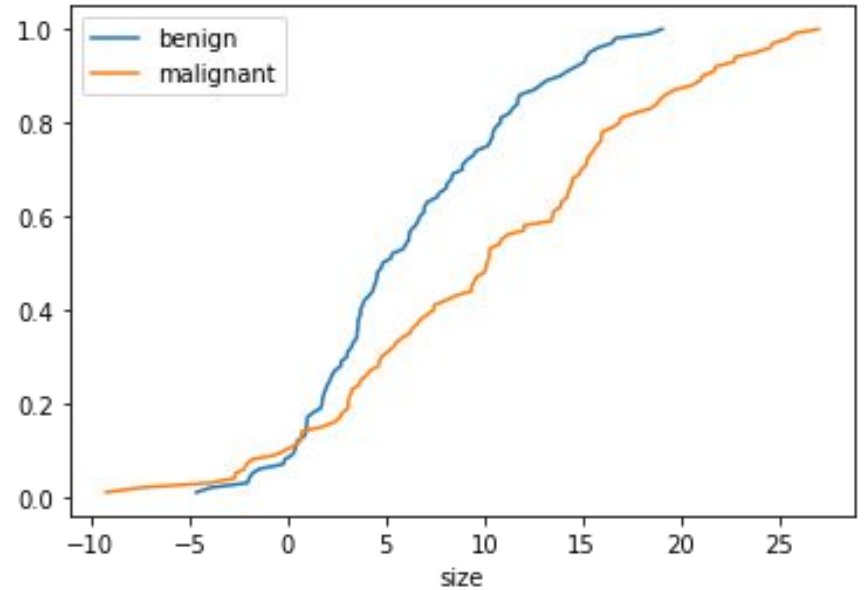
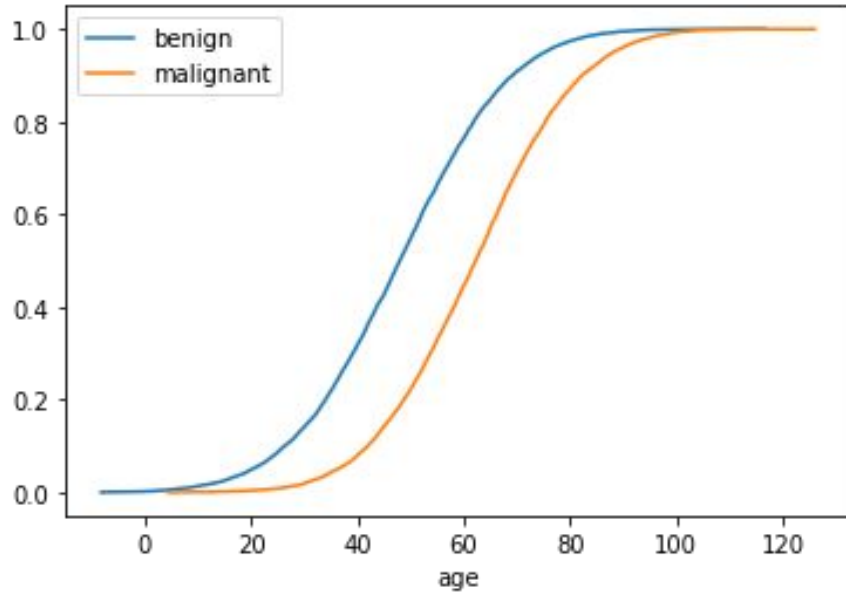
P value of independent T - test : 9.443773241442537e-276

Benign vs. Malignant: Size of the lesion



P value of independent T - test : 2.7279515204510347e-29

ECDF of age range and size of lesion



Location of the lesion and percentage of malignancy

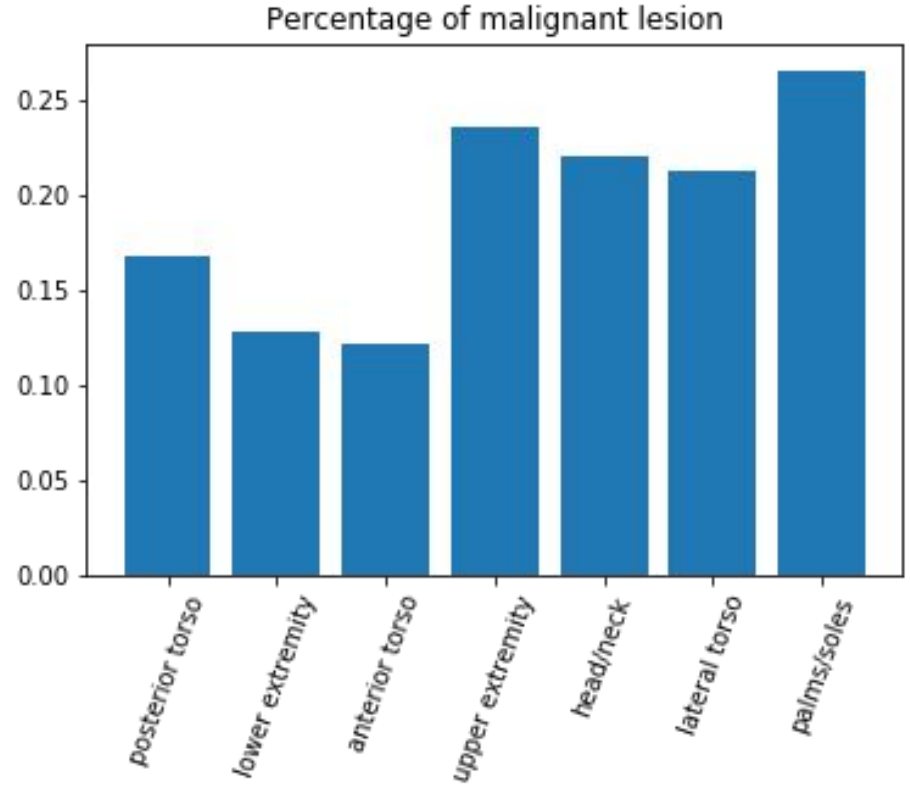
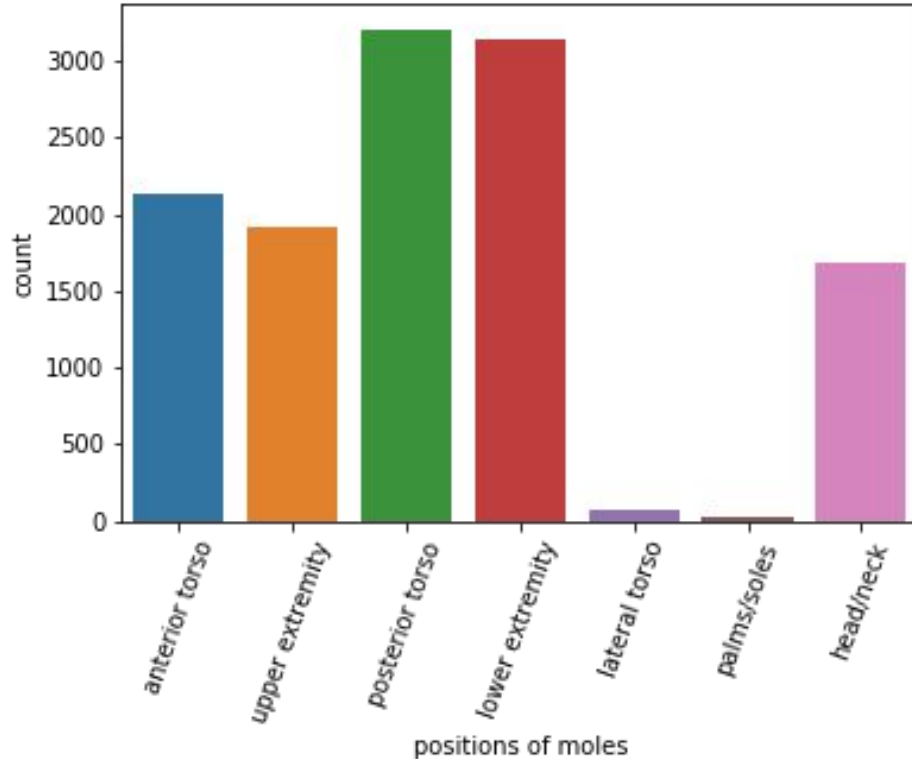
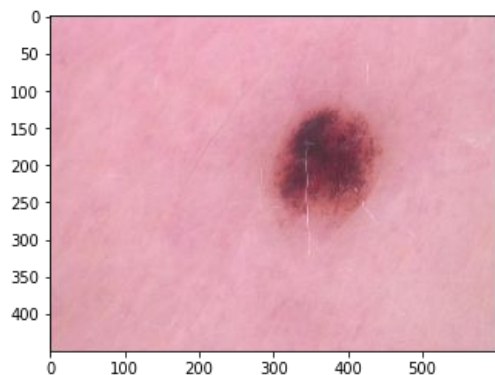
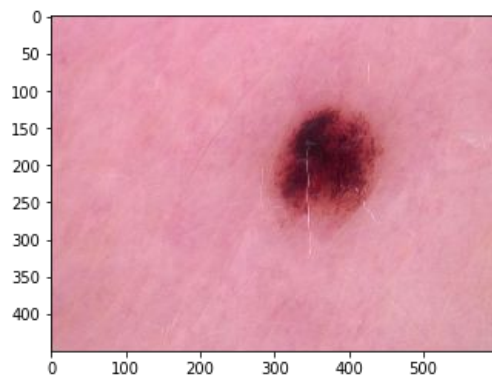


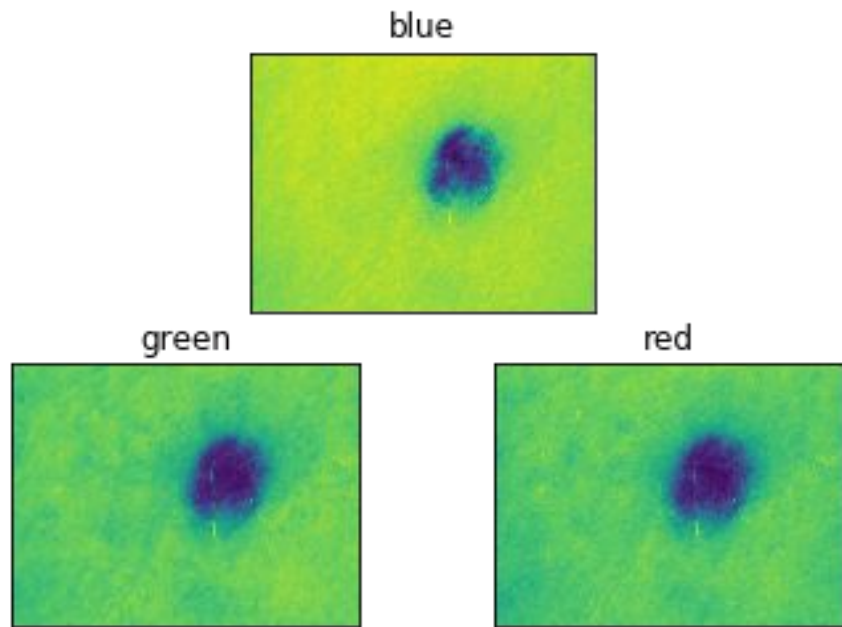
Image preprocessing - normalize and color split



Original



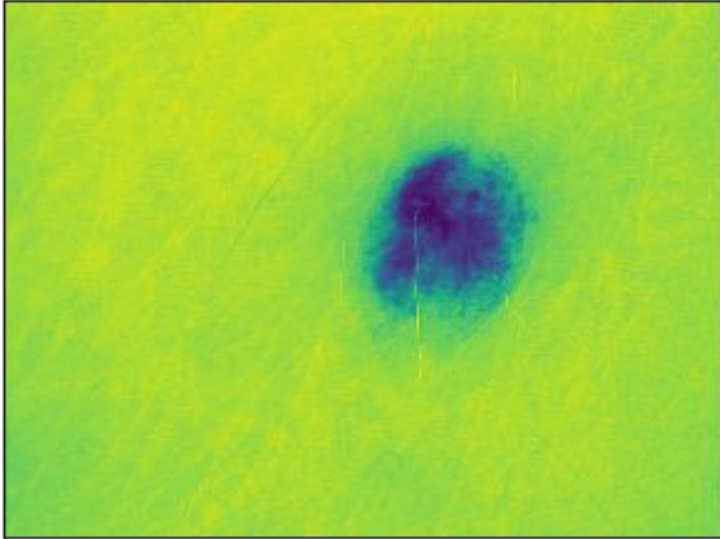
Normalized



Color channel split

Image preprocessing - Gaussian filter

Blue channel



After Gaussian

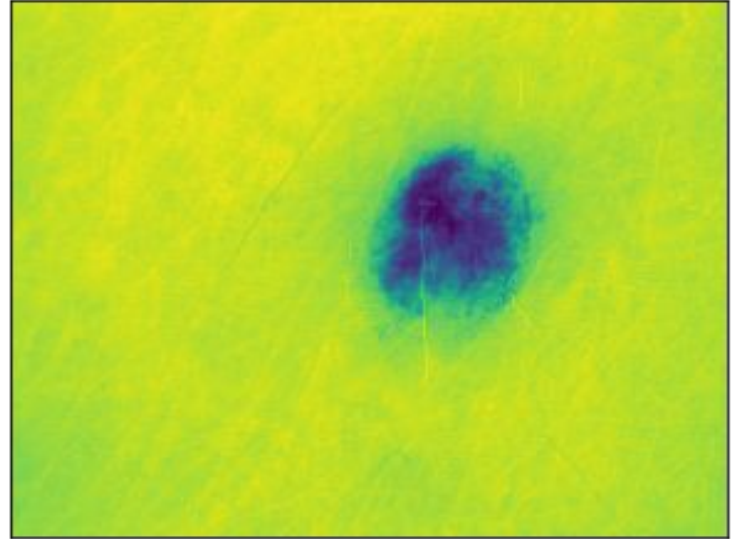
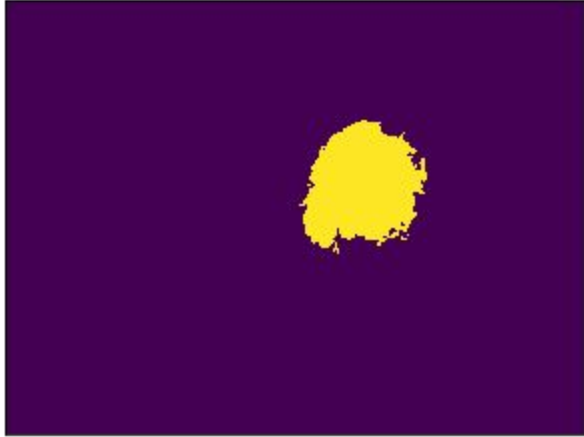


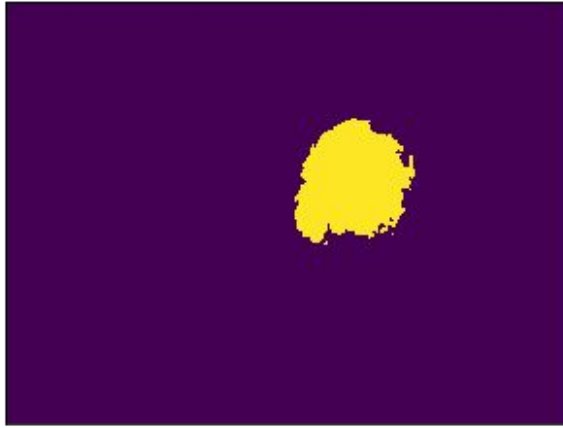
Image preprocessing - Image thresholding

Threshold image



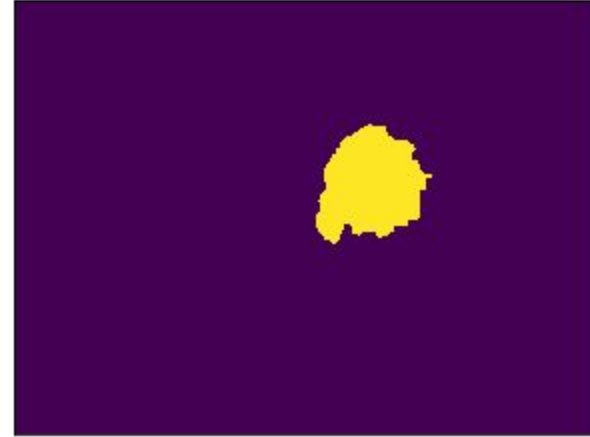
Binary + Otsu's binarization

dilation



Kernel = 3 x 3

erosion



Result of image segmentation

original image



mask on image

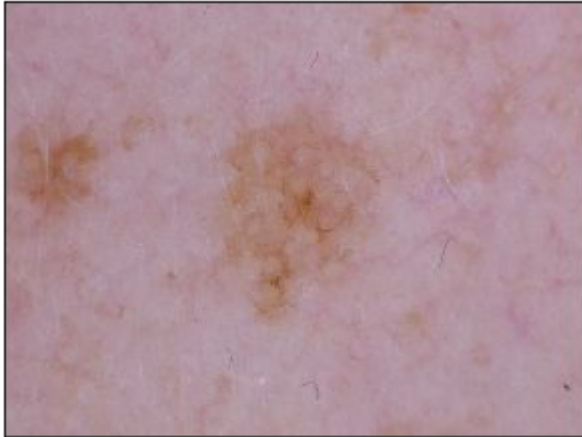


Problem with manual process of image segmentation

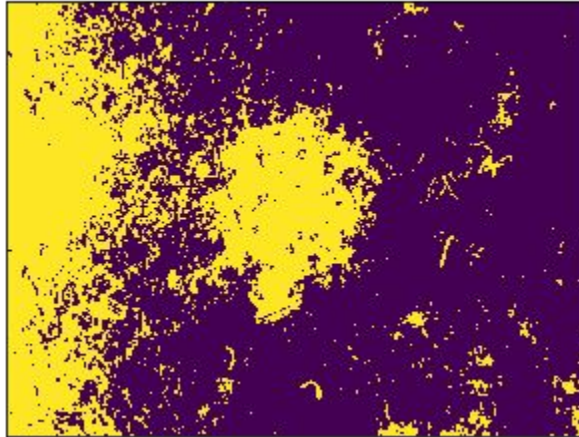
=> Very very slow

=> high chance of running into images that wouldn't work

original image



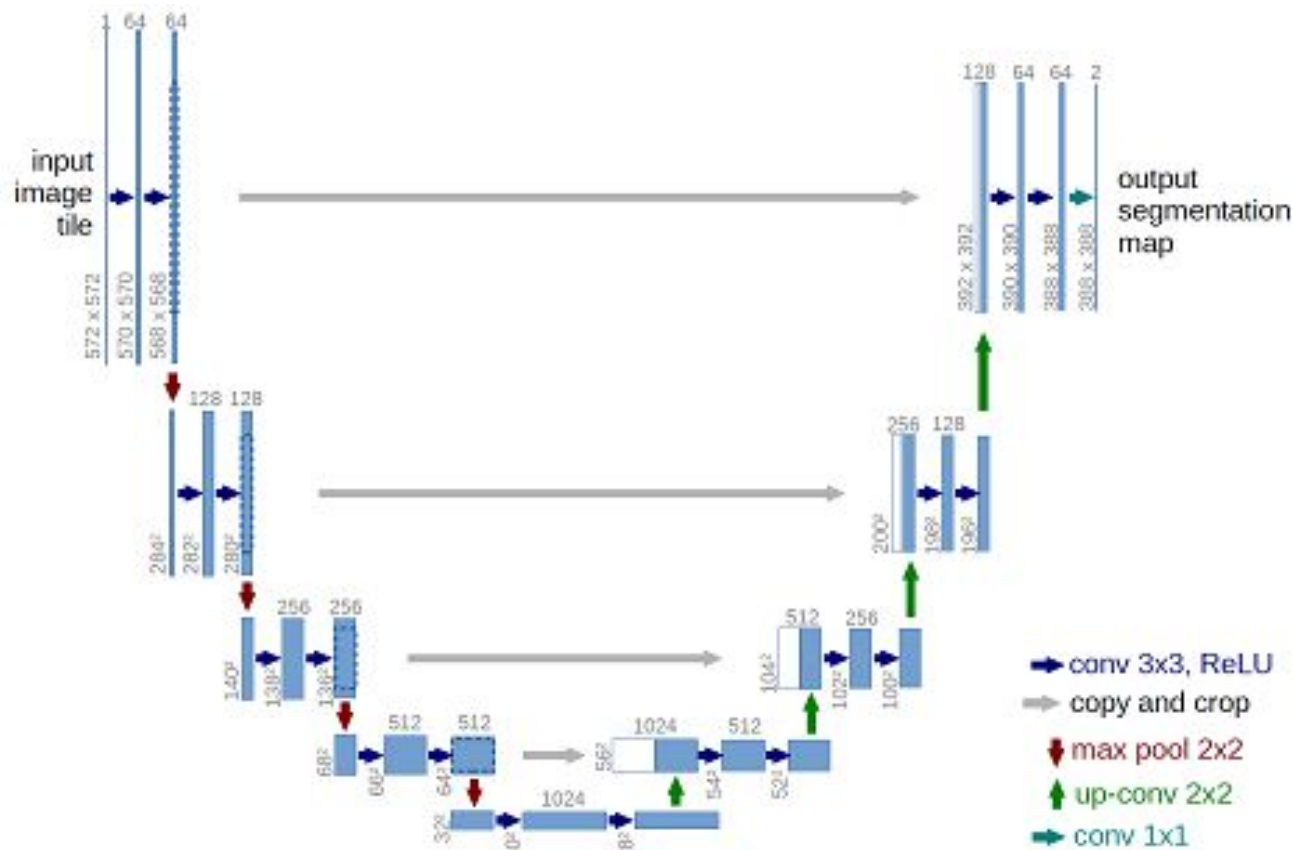
Threshold image



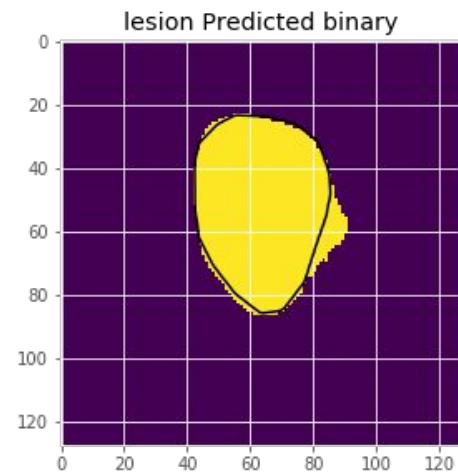
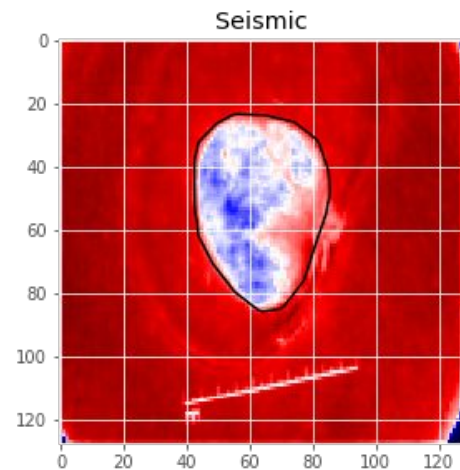
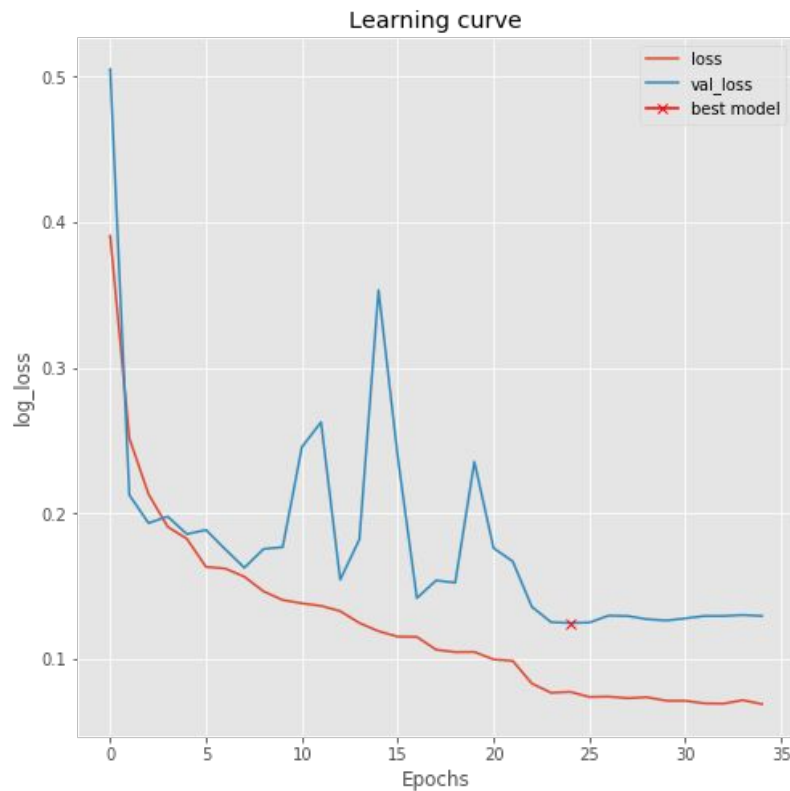
mask on image



Using UNET model for image segmentation



Getting the best model

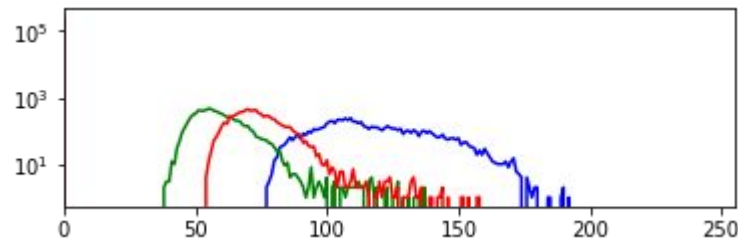
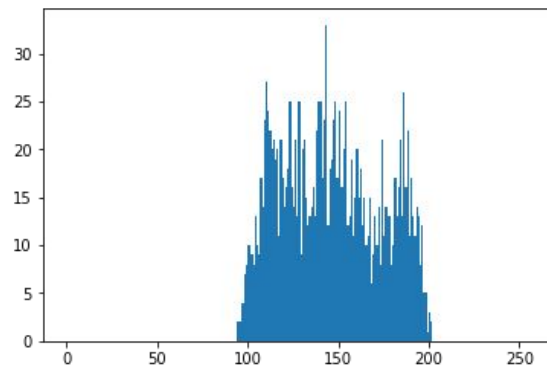
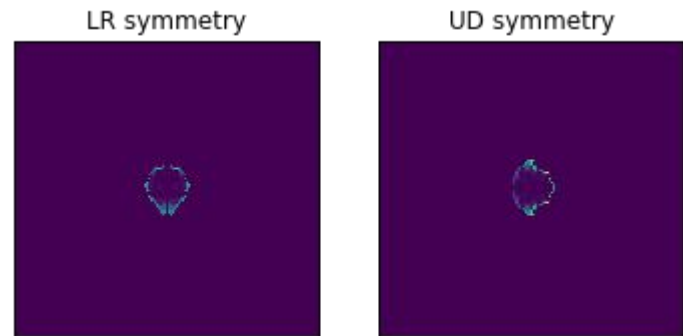


ABC of ABCDE

Asymmetry => tilt the lesion on major axis and compare the difference when flipped horizontally or vertically.

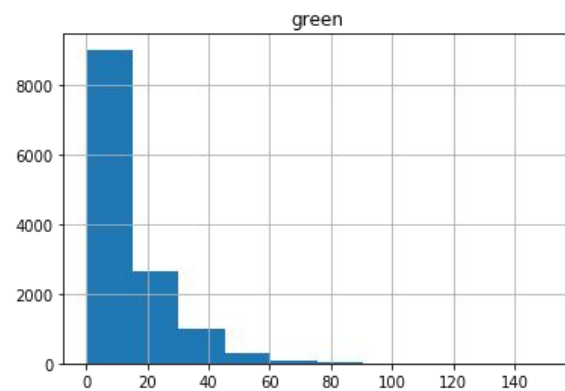
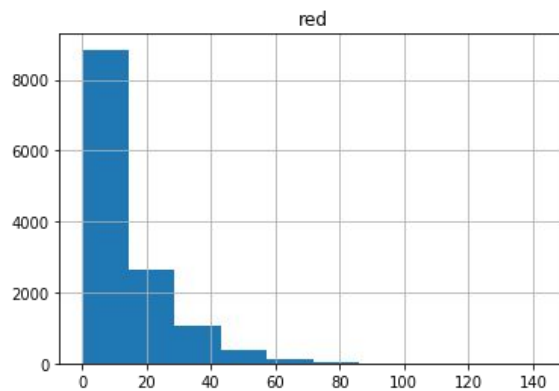
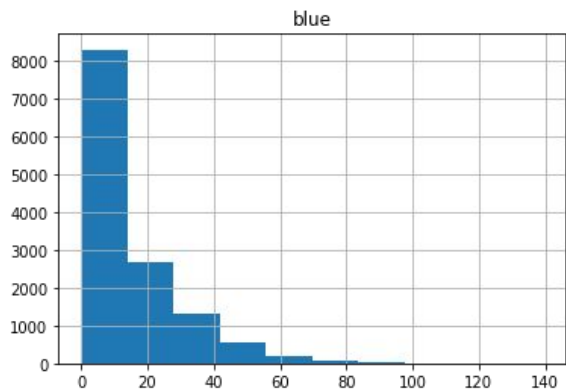
Border=> calculate gradient from cropped image of borders

Color => draw color histogram and get standard deviation

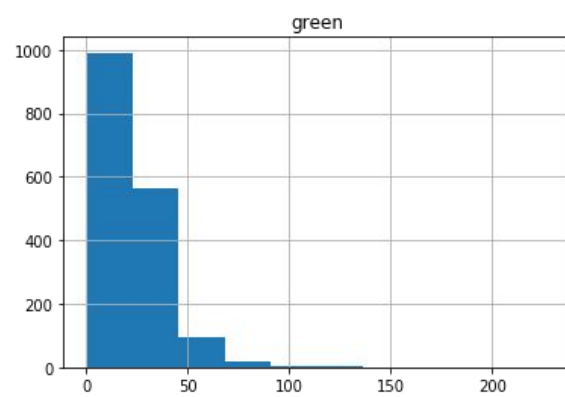
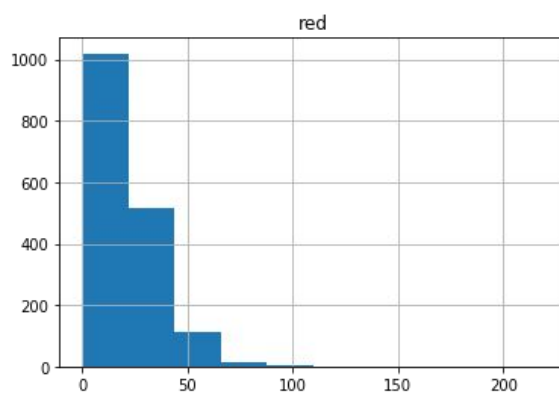
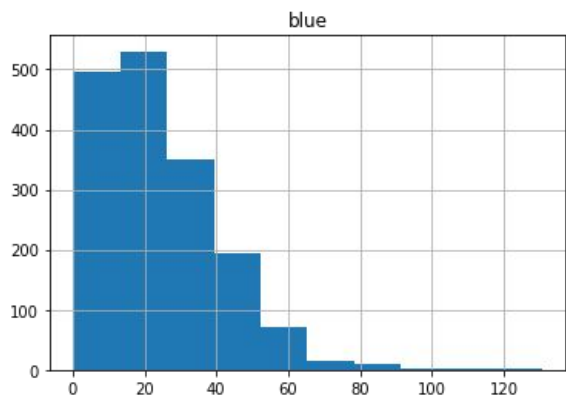


Comparison between benign and malignant lesions

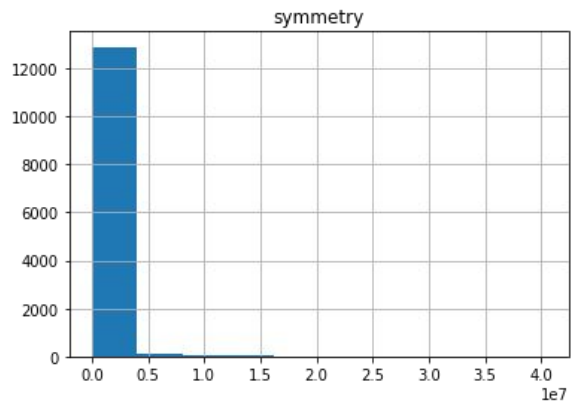
		symmetry	border	red	blue	green
malig	std	3904182.5	0.560375	16.15623	16.63434	16.25676
	mean	815379.11	1.788012	21.0908	24.06696	21.85776
beni	std	1742113.5	0.544875	13.79489	14.79579	13.51443
	mean	241452.07	1.274017	12.59589	14.1669	12.67998



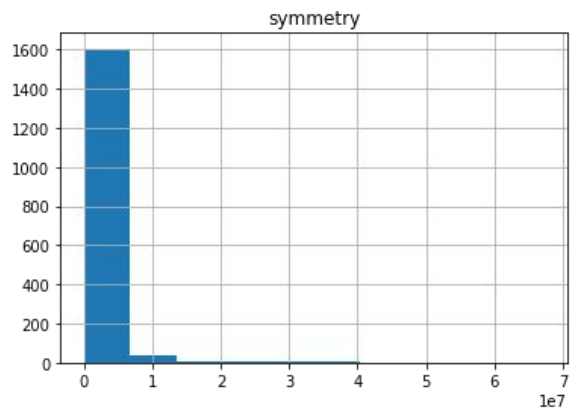
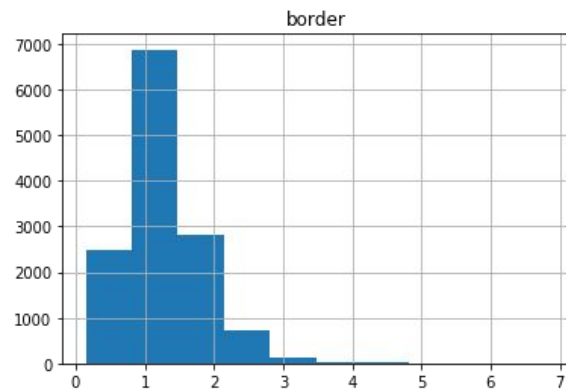
Benign



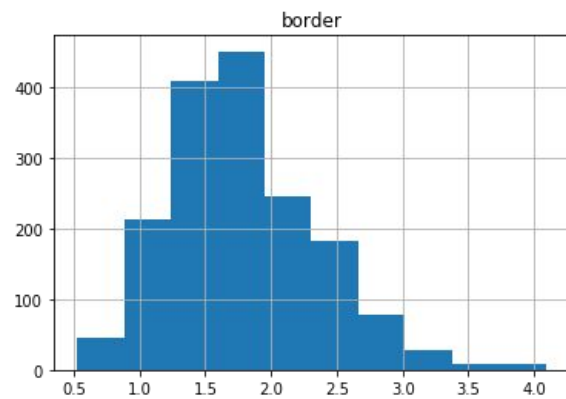
Malignant



Benign



Malignant

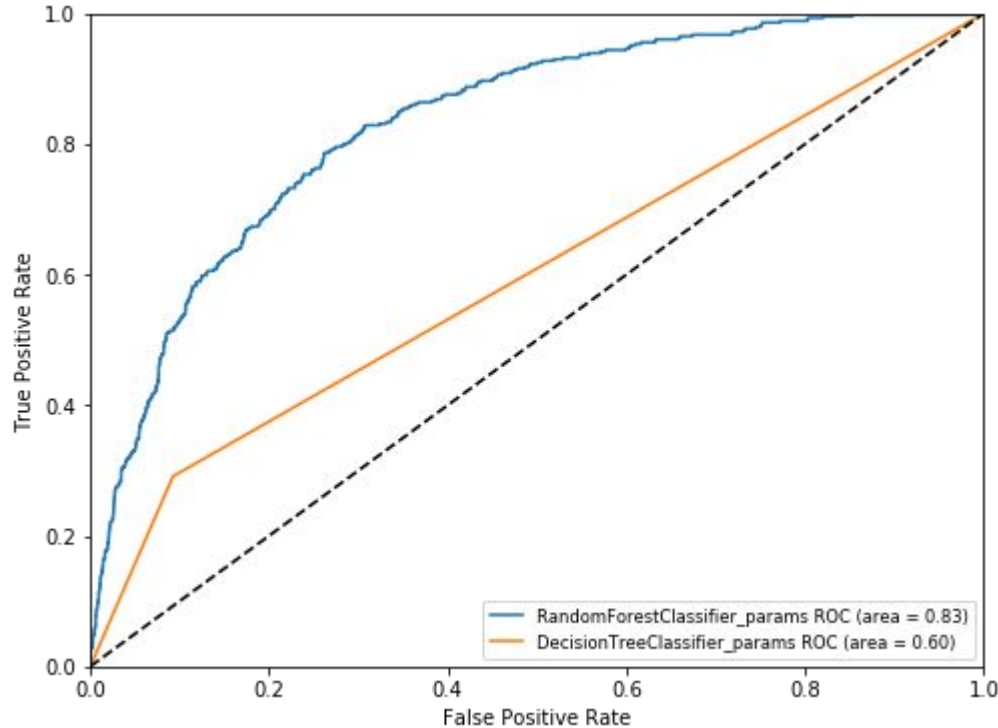


Features used for the model

- Symmetry
- Border
- Blue
- Red
- Green

=> A, B, C of ABCDE rule

Random forest classification vs. Decision tree classification



AUC of Random Forest Classifier 0.83
AUC of Decision Tree Classifier 0.60

Transfer learning

- Retraining the last layer of a pre-trained CNN to classify the images between benign and malignant images
- Faster training
- Does not require much segmentation process

Retraining steps

- Pretrained model: Inception V3 trained on ImageNet
- Training on top layer => training accuracy, validation accuracy, cross entropy
 - Cross entropy => loss function. Should be as small as possible

```
root@aug-27-s-4vcpu-8gb-sfo2-01: ~
6: Step 3980: Train accuracy = 89.0%
I0831 06:00:40.025869 140586214651712 retrain.py:1106] 2019-08-31 06:00:40.02584
3: Step 3980: Cross entropy = 0.298947
I0831 06:00:40.152009 140586214651712 retrain.py:1125] 2019-08-31 06:00:40.15187
5: Step 3980: Validation accuracy = 75.0% (N=100)
I0831 06:00:41.475670 140586214651712 retrain.py:1104] 2019-08-31 06:00:41.47555
2: Step 3990: Train accuracy = 88.0%
I0831 06:00:41.475987 140586214651712 retrain.py:1106] 2019-08-31 06:00:41.47595
7: Step 3990: Cross entropy = 0.314781
I0831 06:00:41.601299 140586214651712 retrain.py:1125] 2019-08-31 06:00:41.60116
8: Step 3990: Validation accuracy = 83.0% (N=100)
I0831 06:00:42.824781 140586214651712 retrain.py:1104] 2019-08-31 06:00:42.82466
7: Step 3999: Train accuracy = 87.0%
I0831 06:00:42.825133 140586214651712 retrain.py:1106] 2019-08-31 06:00:42.82507
4: Step 3999: Cross entropy = 0.318698
I0831 06:00:42.954277 140586214651712 retrain.py:1125] 2019-08-31 06:00:42.95416
6: Step 3999: Validation accuracy = 80.0% (N=100)
2019-08-31 06:00:45.402903: W tensorflow/core/graph/graph_constructor.cc:1352] I
Importing a graph with a lower producer version 29 into an existing graph with pr
ducer version 38. Shape inference will have run different parts of the graph wi
th different producer versions.
I0831 06:00:51.382838 140586214651712 saver.py:1499] Saver not created because t
here are no variables in the graph to restore
W0831 06:00:52.790027 140586214651712 deprecation.py:323] From /home/seo/enviro
```

```
root@aug-27-s-4vcpu-8gb-sfo2-01: ~
I0901 02:40:10.690806 140210782562112 retrain.py:1104] 2019-09-01 02:40:10.69070
3: Step 3980: Train accuracy = 91.0%
I0901 02:40:10.691081 140210782562112 retrain.py:1106] 2019-09-01 02:40:10.69105
6: Step 3980: Cross entropy = 0.180310
I0901 02:40:10.815023 140210782562112 retrain.py:1125] 2019-09-01 02:40:10.81491
7: Step 3980: Validation accuracy = 90.0% (N=100)
I0901 02:40:12.116094 140210782562112 retrain.py:1104] 2019-09-01 02:40:12.11597
4: Step 3990: Train accuracy = 89.0%
I0901 02:40:12.116367 140210782562112 retrain.py:1106] 2019-09-01 02:40:12.11634
2: Step 3990: Cross entropy = 0.230145
I0901 02:40:12.246662 140210782562112 retrain.py:1125] 2019-09-01 02:40:12.24655
7: Step 3990: Validation accuracy = 91.0% (N=100)
I0901 02:40:13.468757 140210782562112 retrain.py:1104] 2019-09-01 02:40:13.46864
6: Step 3999: Train accuracy = 95.0%
I0901 02:40:13.469044 140210782562112 retrain.py:1106] 2019-09-01 02:40:13.46901
8: Step 3999: Cross entropy = 0.115914
I0901 02:40:13.597385 140210782562112 retrain.py:1125] 2019-09-01 02:40:13.59723
4: Step 3999: Validation accuracy = 90.0% (N=100)
2019-09-01 02:40:16.677816: W tensorflow/core/graph/graph_constructor.cc:1352] I
Importing a graph with a lower producer version 29 into an existing graph with pr
ducer version 38. Shape inference will have run different parts of the graph wi
th different producer versions.
I0901 02:40:22.443550 140210782562112 saver.py:1499] Saver not created because t
here are no variables in the graph to restore
```

Resized => cross entropy 0.3187
Validation accuracy 80%

Regular+Resized => cross entropy 0.1160
Validation accuracy 90%

Conclusion

- For the image processing problem, unless we know exactly how each image differs it is useful to use deep learning

More specific conclusion regarding things I tried and found