**7 specifications require changes**

This is a very cool analysis and a great read to a very real world and practical problem. You have demonstrated a full understanding of the entire machine learning pipeline and your report definitely gets the readers attention with the results you have achieved. You just have to expand in some of these sections, but will greatly improve your report. Please check out some of these other ideas presented here and we look forward in seeing your next submission!!

Can also check these out

* (<https://course.fast.ai/>)
* (<https://www.youtube.com/playlist?list=PLkt2uSq6rBVctENoVBg1TpCC7OQi31AlC>)
* (<https://www.youtube.com/playlist?list=PL9Hr9sNUjfsmEu1ZniY0XpHSzl5uihcXZ>)

**Definition**

Student provides a high-level overview of the project in layman’s terms. Background information such as the problem domain, the project origin, and related data sets or input data is given.

Nice work here with your opening section, as you have given good starting paragraphs to outline the project and have provided background information on the problem domain. Definitely a real world problem.

And you have provided good research to back your claims. It is always important to provide similiar research on such a topic.

The problem which needs to be solved is clearly defined. A strategy for solving the problem, including discussion of the expected solution, has been made.

Problem statement is clearly defined here. Just make sure you also provide a brief strategy for solving this problem in this **Problem statement** section. What models will be used? What pre-processing steps? etc... Just a brief overview of your machine learning pipeline.

Metrics used to measure performance of a model or result are clearly defined. Metrics are justified based on the characteristics of the problem.

Good analysis of your metrics. Cohen’s kappa statistic is a cool one to use. Tying your metric choice into your particular problem and problem domain is actually the single most important thing to do in any machine learning project. If you optimize a model based on the incorrect metric, your model might not be suitable for the business goals.

**Analysis**

If a dataset is present, features and calculated statistics relevant to the problem have been reported and discussed, along with a sampling of the data. In lieu of a dataset, a thorough description of the input space or input data has been made. Abnormalities or characteristics about the data or input that need to be addressed have been identified.

Glad that you discuss the size of the dataset and the distribution of the target class. Therefore lastly for this section, make sure you also

* Show some samples images in your report.
* Mention what are the image sizes. All the same size? Varying sizes? etc...
* Show some descriptive stats of some of the most relevant features in the dataset. Maybe something regarding the image sizes?

As this allows the reader to get an understanding of the structure of the data you are working with.

A visualization has been provided that summarizes or extracts a relevant characteristic or feature about the dataset or input data with thorough discussion. Visual cues are clearly defined.

Good visuals for your problem. Maybe also plot some [Intensity Histograms](https://homepages.inf.ed.ac.uk/rbf/HIPR2/histgram.htm). In an image processing context, the histogram of an image normally refers to a histogram of the pixel intensity values. This histogram is a graph showing the number of pixels in an image at each different intensity value found in that image.

Algorithms and techniques used in the project are thoroughly discussed and properly justified based on the characteristics of the problem.

Off to a good start here, however in this Algorithms and Techniques section, you should also go into detail for the internals of a CNN model as well. Since a convolutional neural network main functionality in terms of why it does well with image data is the convolutional layer, you need to give some explanation in how a convolutional layer and pooling layers actually work in detail here. Why are they needed in a CNN model?  
(<http://cs231n.github.io/convolutional-networks/>)

Maybe even some visuals could help explain each step as well.

[](https://udacity-reviews-uploads.s3.us-west-2.amazonaws.com/_attachments/19273/1556338037/Convolution_schematic.gif)

Student clearly defines a benchmark result or threshold for comparing performances of solutions obtained.

External benchmarks are great for such a problem. Glad that you mention their model architecture.

Benchmarking is the process of comparing your result to existing method or running a very simple machine learning model, just to confirm that your problem is actually 'solvable'.

**Methodology**

All preprocessing steps have been clearly documented. Abnormalities or characteristics about the data or input that needed to be addressed have been corrected. If no data preprocessing is necessary, it has been clearly justified.

All the required steps for building a CNN model.

And good idea to use some data augmentation. As this help the model actually 'learn' features and not just memorize images. And also helps generate more training data.

The process for which metrics, algorithms, and techniques were implemented with the given datasets or input data has been thoroughly documented. Complications that occurred during the coding process are discussed.

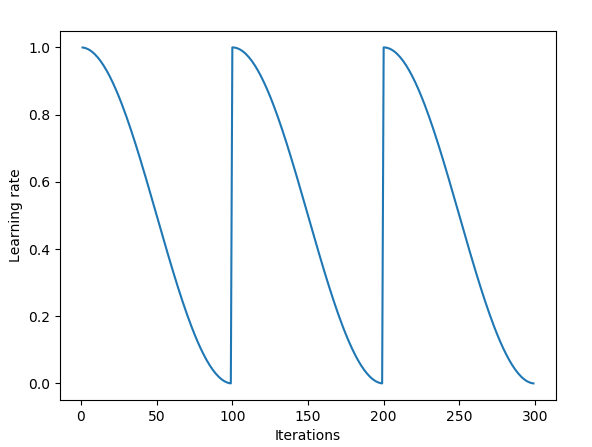
"I have used implementation sections throughout the paper to explain how a topic was implemented in the code."

Ideally, your implementation should be all in one place here. It is a bit tough to navigate your entire paper.

You should describe and document your entire machine learning pipeline here. Describe all the steps you took to code up this algorithm. Think about how someone could replicate your results for this section (without seeing your code).

The process of improving upon the algorithms and techniques used is clearly documented. Both the initial and final solutions are reported, along with intermediate solutions, if necessary.

"I also used a scheduler, lr\_scheduler.reducelronplateau , which allows dynamic learning rate reducing based on validation measurements."

Good idea! Another idea would be to check out using [Cyclical Learning Rates for Training Neural Networks](https://arxiv.org/abs/1506.01186). This is where we simply keep increasing the learning rate from a very small value, until the loss stops decreasing and then bump it up once more. We can plot the learning rate across batches to see what this looks like.  
[](https://udacity-reviews-uploads.s3.us-west-2.amazonaws.com/_attachments/19273/1519240063/CLR_plot.png)

**Results**

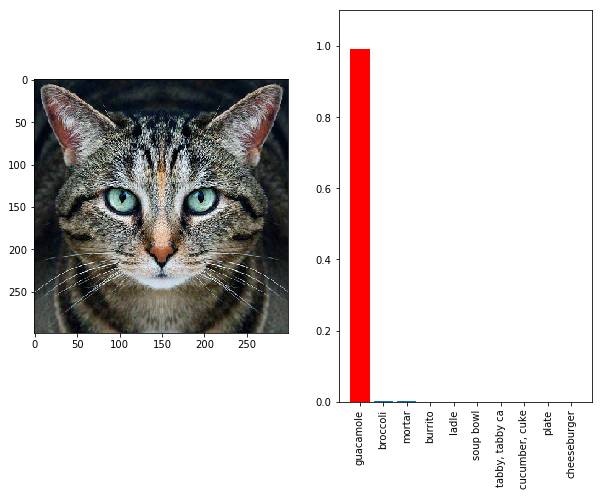
The final model’s qualities — such as parameters — are evaluated in detail. Some type of analysis is used to validate the robustness of the model’s solution.

You have done a good job describing your final model, however also for this section, you need to also provide some discussion to validate the robustness of the model’s solution.

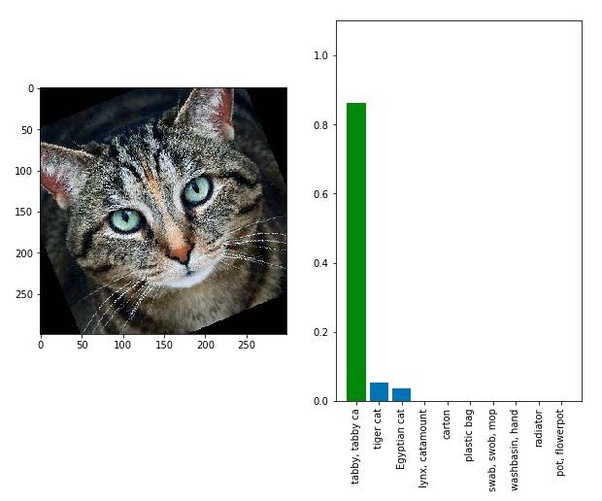
Some ideas to discuss / do to validate your model

* Could look into using KFold CV and show the different folds scores
* Go into more detail in terms of your training / validation results. Any overfitting happening here? Do we need more regularization? dropout? etc... Would you consider the model 'robust' based on your visual? Why?
* Another idea to validate the robustness of the model’s solution, would be to try and add some random gaussian noise the images and see how the model performs? Did it actually learn feature or is it just memorizing images?

**Why we need to validate our models?**

[](https://udacity-reviews-uploads.s3.us-west-2.amazonaws.com/_attachments/19273/1533482589/cat1.jpeg)

Here’s a picture of a cat right? Google’s Inception model thinks it’s a guacamole. As much as the image looks like a cat, the image is digitally altered which confused the model.

[](https://udacity-reviews-uploads.s3.us-west-2.amazonaws.com/_attachments/19273/1533482606/cat2.jpeg)

Slightly rotating the image led the model to correctly classify the image as a cat (and as an animal)

The above image is what’s called as an adversarial image, trying to fool your model into thinking the image is something you want it to be instead of what the image actually is.

The real danger is in the application, especially in healthcare and defense. For example, how would you convince that your model for predicting cancer actually works? How do you know your model is not susceptible to noise? How do you know that your model has actually learnt what it is supposed to be learning? This is why it is always important to validate the robustness of the model’s solution.

The final results are compared to the benchmark result or threshold with some type of statistical analysis. Justification is made as to whether the final model and solution is significant enough to have adequately solved the problem.

**Conclusion**

A visualization has been provided that emphasizes an important quality about the project with thorough discussion. Visual cues are clearly defined.

Please include a **Free-Form Visualization** section. In this section, you will need to provide some form of visualization that emphasizes an important quality about the project. It is much more free-form, but should reasonably support a significant result or characteristic about the problem that you want to discuss.

Student adequately summarizes the end-to-end problem solution and discusses one or two particular aspects of the project they found interesting or difficult.

Please include a **Reflection** section. In this section, you will summarize the entire end-to-end problem solution and discuss one or two particular aspects of the project you found interesting or difficult. You are expected to reflect on the project as a whole to show that you have a firm understanding of the entire process employed in your work.

Discussion is made as to how one aspect of the implementation could be improved. Potential solutions resulting from these improvements are considered and compared/contrasted to the current solution.

Please include an **Improvement** section. In this section, you will need to provide discussion as to how one aspect of the implementation you designed could be improved. As an example, consider ways your implementation can be made more general, and what would need to be modified. You do not need to make this improvement, but the potential solutions resulting from these changes are considered and compared/contrasted to your current solution.

**Quality**

Project report follows a well-organized structure and would be readily understood by its intended audience. Each section is written in a clear, concise and specific manner. Few grammatical and spelling mistakes are present. All resources used to complete the project are cited and referenced.

Your **Implementation** sections throughout are a bit tough to follow. Would recommend describing your entire implementation solely in your **Implementation** section.

Code is formatted neatly with comments that effectively explain complex implementations. Output produces similar results and solutions as to those discussed in the project.

Looks good! You might also check out

* this [post](https://stackoverflow.com/questions/19074745/docstrings-vs-comments) regarding Docstrings vs Comments.
* [Google Style Python Docstrings](https://sphinxcontrib-napoleon.readthedocs.io/en/latest/example_google.html)
* This [Best of the Best Practices" (BOBP) guide to developing in Python](https://gist.github.com/sloria/7001839)