Snowy Road Classification with a Data-Centric Approach

Brian Hill *bwhill@stanford.edu*, Li Tian lii@stanford.edu, & Joanne Zhou *joannezhou@stanford.edu*

Background & Problem Statement

After a winter trip to Alaska, we were inspired to develop and apply computer vision architectures to recognize winter road conditions to improve driving safety.

The difficulty of the classification problem is multi-fold. The ideal model should not only recognize snow coverage in a variety of road context, but also differentiate the levels of coverage, during both daytime and nighttime. Traditional image recognition technology currently cannot achieve the fast real-time high-accuracy performance necessary for road recognition in intelligent and safe driving, but deep learning models have emerged as promising tools to achieve this performance, yet most of the architectures currently applied to this problem are outdated by modern standards.

Dataset

Location:
South-Western Ontario, Canada

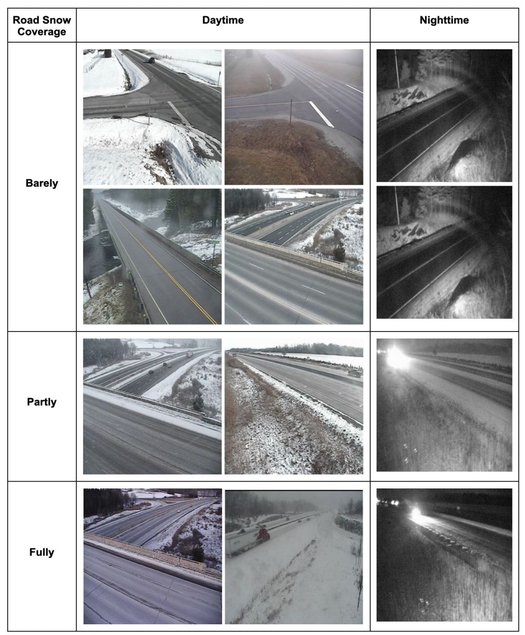
Snowfall:
59 days/year (0.2cm+)

Station count:
60 Stations

Train-Val-Test Split:
70%-15%-15%

Size:
21,540 Images

Source:
Prof. Pan (Linyi University)

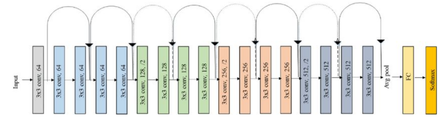


In previous works, Pan et. al separated all images at random to form their splits. However, we believe this technique would result in data leakage, since similar images from the same camera station have likely already been seen during training. Thus, we decided to split the data set by station, such that the validation and test sets contain a similar distribution of different classes, but from an unseen camera perspective. This will make the classification task more challenging, but more realistic and generalizable to newly installed camera stations that have no labeled data available.

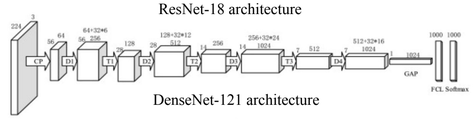
Snow Coverage	Image count	Percentage
Barely	9,506	44.1%
Partly	8,930	41.5%
Fully	3,104	14.4%
Total	21,540	100%

Methods

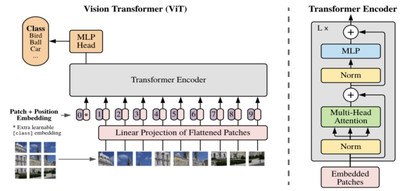
++++ Model-Centric :Model Selection ++++



ResNet-18 architecture



DenseNet-121 architecture



Vision Transformer architecture

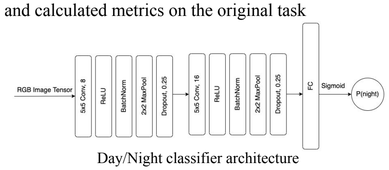
Models	Architecture Characteristics & Advantages	Variations
ResNet	Identity mapping in the residual block ensures that the model takes the representation from earlier layers as an input in following steps -> enables training of deeper models	ResNet-18 ResNet-34
DenseNet	Dense blocks uses the feature-maps of all preceding and its own feature-maps as inputs -> strengthening feature propagation and encouraging feature reuse with small dataset.	DenseNet-121 DenseNet-161
Vision Transformer	Multihead attention blocks and position encoding -> efficient training of long sequence and parallel computing.	Vit-16 Vit-32

++++ Data-Centric: Subclass Experiment ++++

Would it be helpful for our model's ability to learn if we separate each coverage type by time of day to create more uniform classes?

We hypothesize that this could help our model learn the task since otherwise two nighttime pictures looks much more similar, even for a human, than different snow coverage levels at the same time of day. To do this our method was:

1) Manually label a small day/night dataset
2) Train a CNN model to predict labels for all images
3) Concatenate day/night and snow coverage labels to create subclasses. Train six-class classifier models with subclasses and calculated metrics on the original task



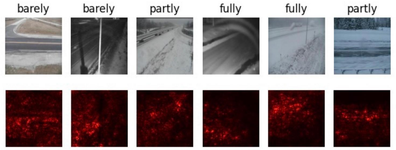
Day/Night classifier architecture

Results

++++ Model Selection ++++

Takeaways:

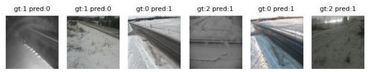
1) Across all models, fine-tuning more layers results in better accuracy.
2) Saliency maps shows that the model was able to focus on key parts of the images for classification (road surface, etc)
3) Three different model architectures achieve similar test accuracy after hyperparameter search and fine-tuning. More complicated model does not guarantee substantial model improvement.



Saliency maps samples for each class

Test Set Results

Model	Fine-tune	Acc
ResNet-18	1	0.7760
ResNet-18	3	0.8190
ResNet-18	all	0.8385
ResNet-18	not pre-trained	0.7370
ResNet-34	all	0.8335
DenseNet-121	1	0.7605
DenseNet-121	**all**	**0.8518**
DenseNet-121	not pre-trained	0.7556
DenseNet-161	all	0.8082
ViT-L/16	all	0.8270
ViT-L/32	all	0.8165
ViT-B/16	all	0.8499
ViT-B/32	all	0.8143



Misclassification samples for each class
0: barely, 1: partly, 2: fully

++++ Subclass Experiment ++++

CIFAR-10 Experiment

When considering non-animal vs. animal classes, little improvement was gained from training on the original 10-class task.

Test Set Results

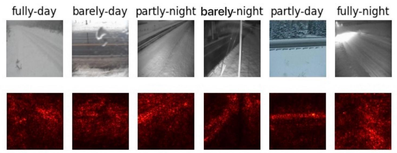
2-classes	0.8742
10-classes Sum	0.8736
10-classes Max	0.8682

Snowy Road Dataset

1) Overall, minimal differences in performance
2) Better recall for classes we are most concerned about identifying: partial and fully snowy
3) Saliency maps tend to focus on sky, misclassifications look similar overall

Subclass Test Results

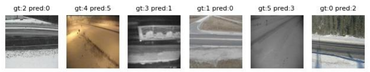
Model	Acc
ResNet-18	0.8382
ResNet-34	0.8363
DenseNet-121	0.8502
DenseNet-161	**0.853**
ViT-B/16	0.7438



Saliency maps samples for each subclass

Recall by Model and Road Condition

	Barely	Partly	Fully
3-Class	**0.9009**	0.8315	0.7747
Subclass	0.8728	**0.8508**	**0.8077**



Misclassification samples for each subclass
0: barely-day, 1: barely-night, 2: partly-day, 3: partly-night, 4: fully-day, 5: fully-night.

Conclusions

Improved Task Performance

Our best performing model achieves good test accuracy (0.85) when compared to previous work on this dataset (0.87), even though we split the data in a way that made for a more challenging task. When splitting data at random as Pan did we achieved accuracies of 0.94, though we do not highlight these results as the test set is certainly polluted.

Model-Centric → Data-Centric

Using deeper models with more parameters does not necessarily improve results significantly because the less complicated models might have already reached the learning capacity, indicated by the high train accuracy.

Subclasses Demonstrate Model Power

This approach was not significantly helpful in improving the test accuracy, potentially because models have already learned the latent classes without being given explicit labels. However, the subclass approach results in better recall rate for the partly and fully snow coverage classes.

Future Directions

Improving the dataset by applying additional data-centric approaches, such as mislabel identification, augmentation, influence functions (which other images influence the prediction of a given image), as well as better understanding how the model learns subclasses by fine-tuning the 6 class model to 3 classes.