Weather Image Recognition

W207 Summer 2023 - Final Project

Jason Rudianto | Savinay Chandrupatla | Xin Chen

Berkeley
UNIVERSITY OF CALIFORNIA

Team Member and Contribution

- Jason Rudianto
  - Data loading/preprocessing , baseline models , overall discussions, presentation slides
- Savinay Chandrupatla
  - CNN modeling and fine tuning, overall discussions, presentation slides
- Xin Chen
  - EDA and baseline models, overall discussions, presentation slides ,NeurIPS checklist

Purpose

The purpose of this research is to perform an end to end dataset analysis on weather images to study the ‘best’ machine learning model to predict the weather type given an image

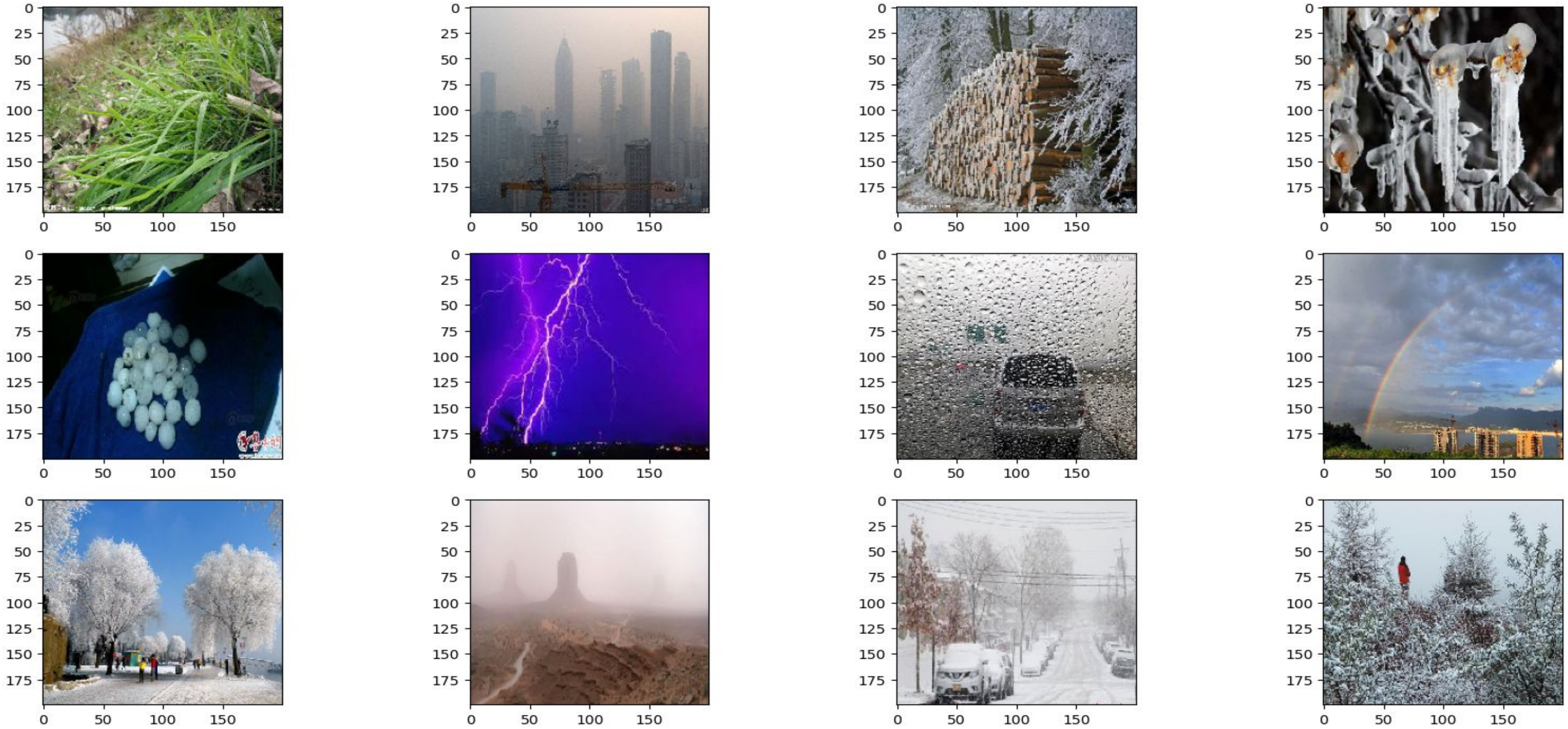
About the data

Source : https://www.kaggle.com/datasets/jehanbhathena/weather-dataset

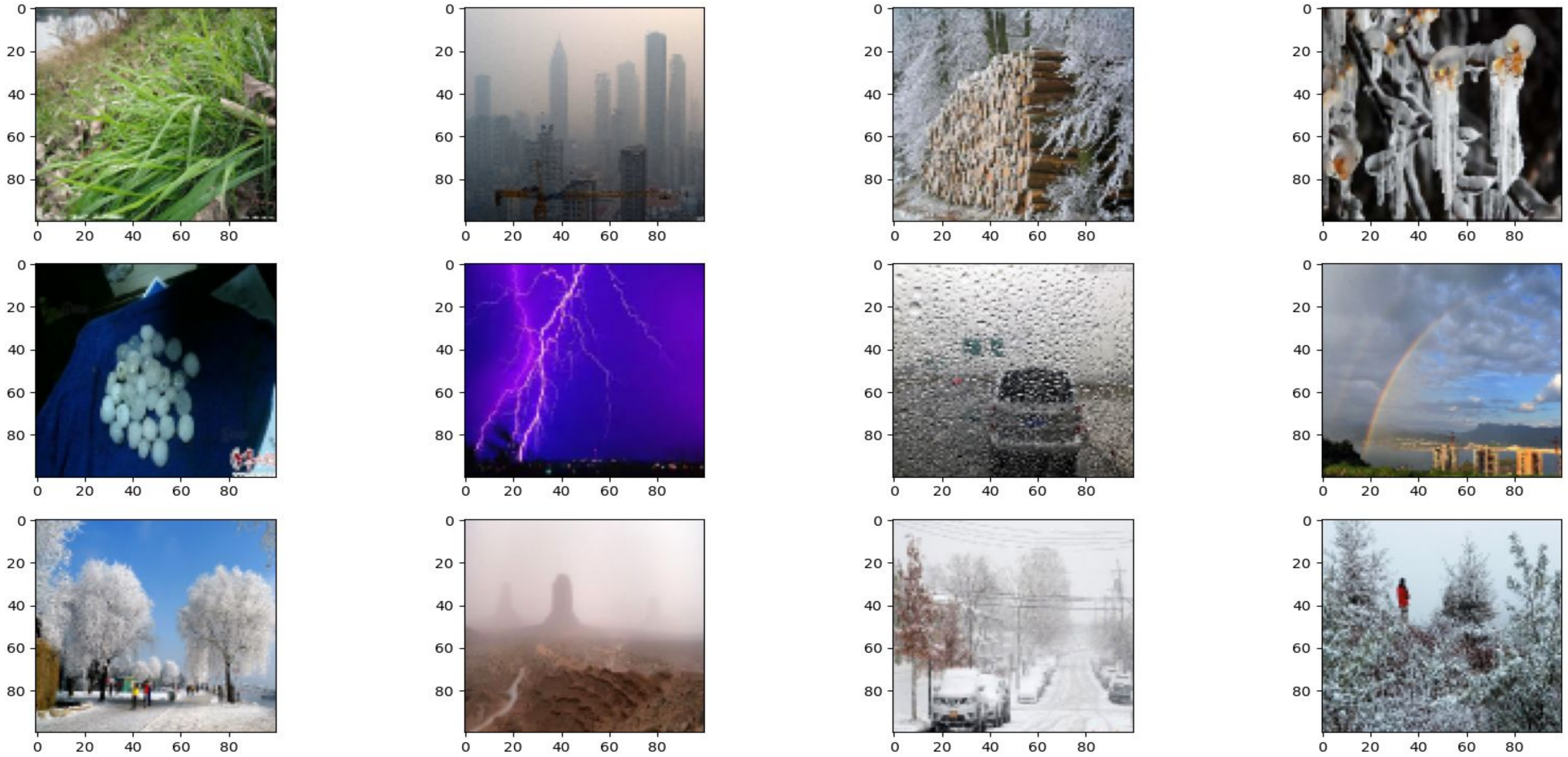
- Contains 6862 images of different types of weather
- 11 weather classification :dew, fog/smog, frost, glaze, hail, lightning , rain, rainbow, rime, sandstorm and snow
- Dataset originally collected for study “Classification of Weather Phenomenon From Images by Using Deep Convolutional Neural Network. Earth and Space Science, 2021” - Xiao H , Zhang F , Shen Z , et al.
    - “In this paper, we initially collected weather phenomena JPG images from the internet and academic exchanges, then manually labeled weather phenomena images using meteorological criteria.”
- Raw images are not standardized – they vary in dimension, resolution and source

Berkeley
UNIVERSITY OF CALIFORNIA

About the data: Size = [200, 200]



About the data: Size = [100, 100]



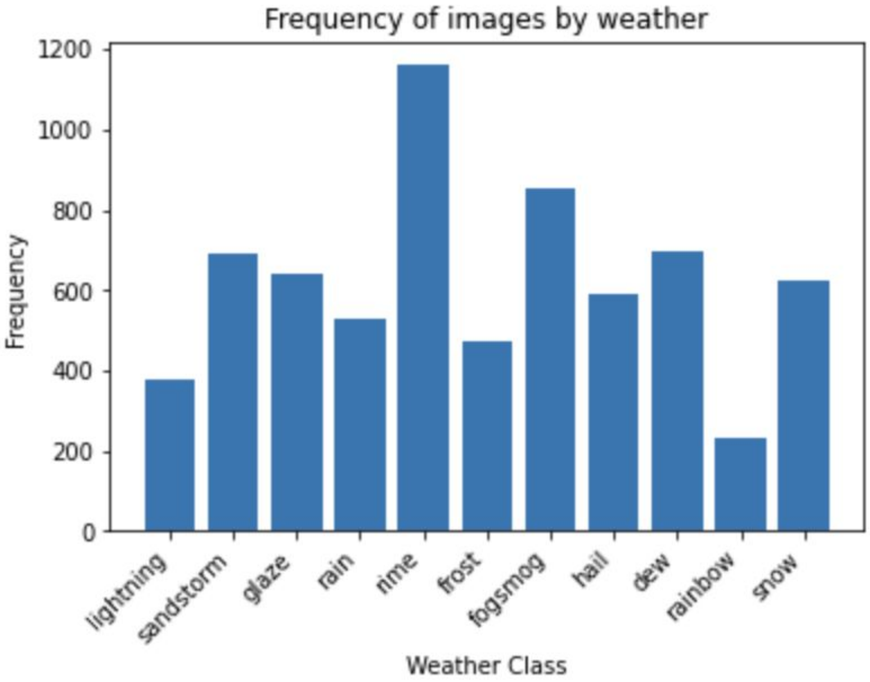
Data Preprocessing

- Loading dataset and extracting y label for directory structure
    - dataset/
        - dew/
            - img_123.png
            - ...
        - snow/
            - image_235.png
            - ...
        - .../
- Normalize RGB values
- Standardize image dimensions
- Augmented additional images (flip, contrast)
- Sample equal amounts of images for training, validation and testing

Berkeley
UNIVERSITY OF CALIFORNIA

Exploratory Data Analysis

- Data imbalance between weather classes
- Random sample at most N=200 entries per class for training, validation and testing
- Motivation: ensure each class is trained on equally



Berkeley
UNIVERSITY OF CALIFORNIA

Exploratory Data Analysis

- Check any color channel differences between weather types

	Color Channel		
	R	G	B
lightning	0.438565	0.440829	0.575204
sandstorm	0.792902	0.657739	0.510932
glaze	0.629989	0.628475	0.614101
rain	0.629603	0.633664	0.613480
rime	0.726930	0.783131	0.863588
frost	0.611385	0.600778	0.573250
fogsmog	0.752962	0.755081	0.753742
hail	0.668579	0.656150	0.610651
dew	0.514041	0.627008	0.419371
rainbow	0.664843	0.708333	0.764998
snow	0.776749	0.781515	0.795303

Berkeley UNIVERSITY OF CALIFORNIA

Approach and Models

Hypothesis: CNN models are better than linear models for image recognition and are also better than NN models for dealing with position invariance

Baseline models

- KNN
- Multi-class logistic regression model
- Multi-class neural network (NN)
  - Fine tuning NN

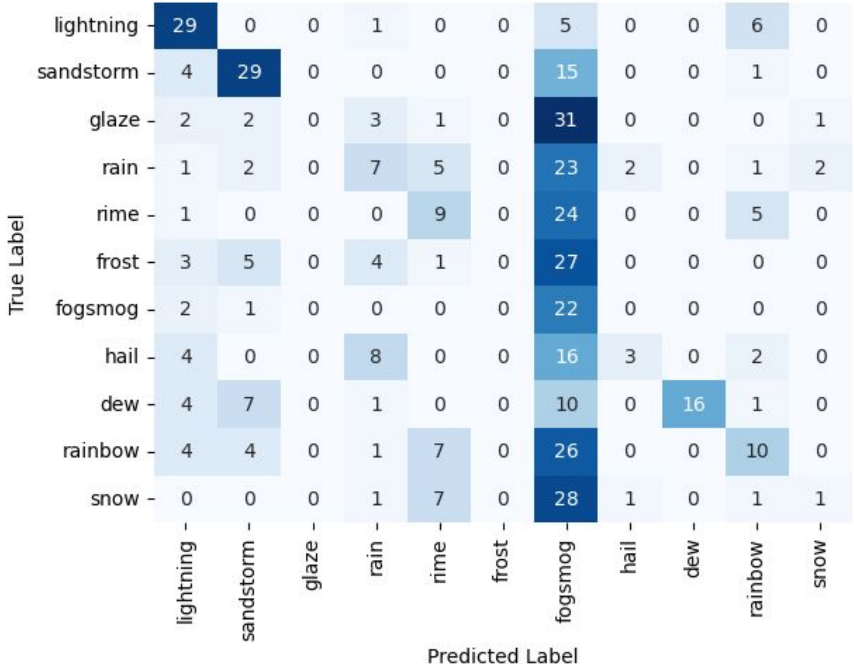
Best choice model as our hypothesis

- CNN
  - Fine tuning and selecting the best CNN model

Berkeley
UNIVERSITY OF CALIFORNIA

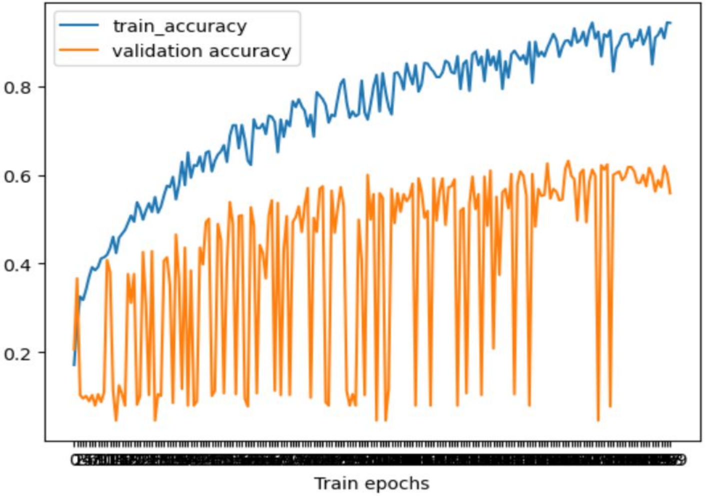
KNN

- Validation accuracy : 0.3
- Test accuracy: 0.29
- Lack of ability to handle position invariance
- Prediction very concentrated
- Not a good baseline model



Baseline model: multi-class logistic regression

- Image dimensions: 200 X 200
- Observations :
    - Unstable val performance
    - Stability improved w/more epochs
- A reasonably good baseline model with accuracy .4




```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 flatten (Flatten)           (None, 120000)            0

 dense (Dense)               (None, 11)                1320011

=================================================================
Total params: 1,320,011
Trainable params: 1,320,011
Non-trainable params: 0
_________________________________________________________________
None
Test Accuracy: 0.4081
```

Baseline model: multi-class logistic regression

- Performance: confusion matrix
  - Better precision in lightning, sandstorm, rain, rime and dew weather types
  - Worst in predicting fog smog
    - Often mistaken as sandstorm

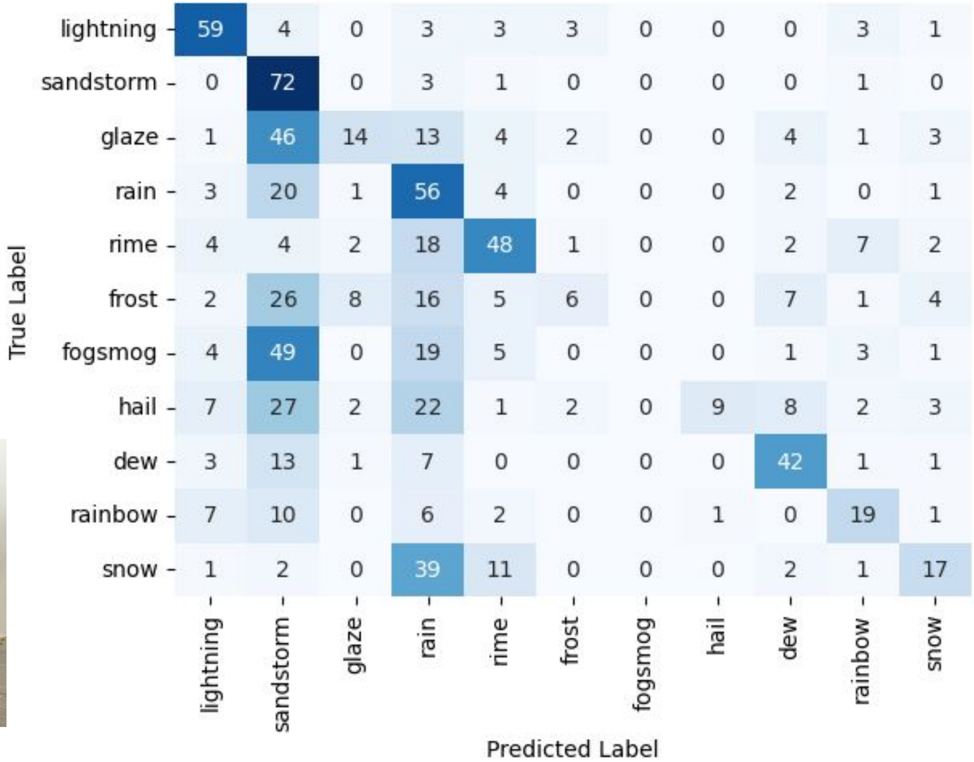



Image of fog smog


Image of Sandstorm

Berkeley
UNIVERSITY OF CALIFORNIA

Baseline model: multi-class neural network(NN)

IMAGE_DIMENSION=(200,200)					
HIDDEN SIZES	ACTIVATION	OPTIMIZER	LEARNING RATE	#PARAMETERS	TEST ACCURACY
[64]	relu	Adam	0.01	7680779	0.0866
[256]	relu	Adam	0.01	30723083	0.2064
[256, 128]	relu	Adam	0.01	30754571	0.2159
[256, 128, 64	relu	Adam	0.01	30762123	0.1684

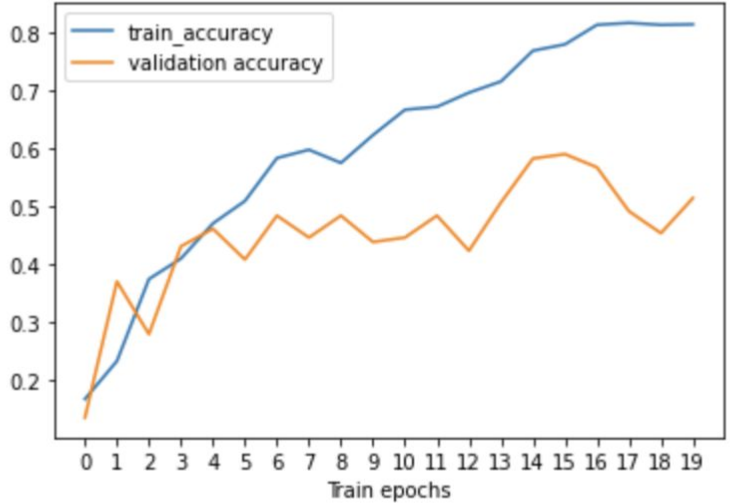
IMAGE_DIMENSION=(100,100)					
HIDDEN SIZES	ACTIVATION	OPTIMIZER	LEARNING RATE	#PARAMETERS	TEST ACCURACY
[64]	relu	Adam	0.01	1920779	0.1435
[256]	relu	Adam	0.01	7683083	0.2325
[256, 128]	relu	Adam	0.01	7714571	0.1815
[256, 128, 64	relu	Adam	0.01	7722123	0.2171

Berkeley
UNIVERSITY OF CALIFORNIA

Baseline Model: NN with additional tuning

- Image dimensions: 32x32 (smaller)
- 1 hidden layer of size 1024
- Adam Optimizer with learning rate 0.001

```
Training...
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 flatten (Flatten)           (None, 3072)              0

 dense (Dense)               (None, 1024)              3146752

 dense_1 (Dense)             (None, 11)                11275

=================================================================
Total params: 3158027 (12.05 MB)
Trainable params: 3158027 (12.05 MB)
Non-trainable params: 0 (0.00 Byte)
_________________________________________________________________
```



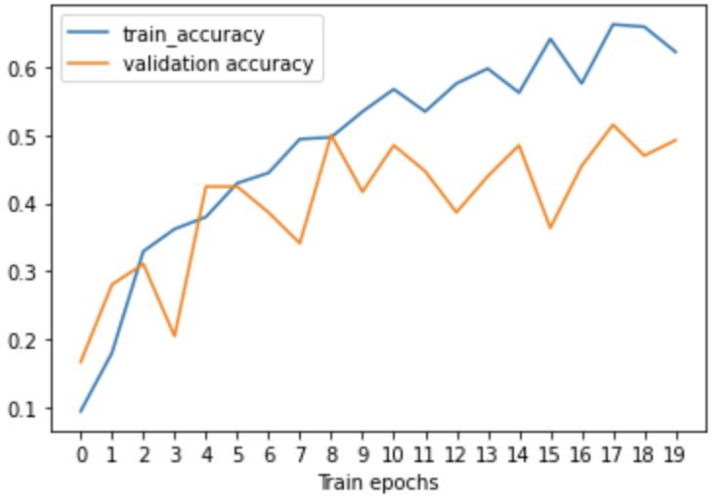

```
Test Accuracy: 0.4045
```

Berkeley
UNIVERSITY OF CALIFORNIA

Baseline Model: NN with additional tuning

- Image dimensions: 32x32 (smaller)
- 3 hidden layers of size [1024,512, 256]
- Adam Optimizer with learning rate 0.001

```
Training...
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 flatten (Flatten)           (None, 3072)              0

 dense (Dense)               (None, 1024)              3146752

 dense_1 (Dense)             (None, 512)               524800

 dense_2 (Dense)             (None, 256)               131328

 dense_3 (Dense)             (None, 11)                2827

=================================================================
Total params: 3805707 (14.52 MB)
Trainable params: 3805707 (14.52 MB)
Non-trainable params: 0 (0.00 Byte)
_________________________________________________________________
```



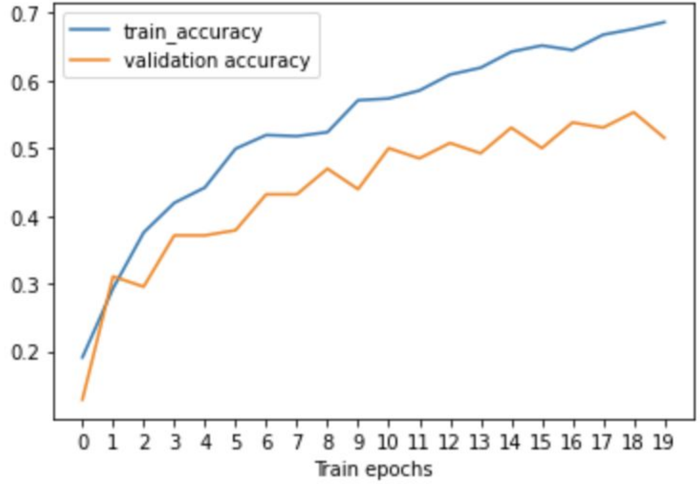

```
Test Accuracy: 0.4318
```

Berkeley
UNIVERSITY OF CALIFORNIA

Baseline Model: NN with additional tuning

- Image dimensions: 32x32 (smaller)
- 3 hidden layers of size [1024,512, 256]
- Adam Optimizer with learning rate 0.00001

```
Training...
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 flatten (Flatten)           (None, 3072)              0

 dense (Dense)               (None, 1024)              3146752

 dense_1 (Dense)             (None, 512)               524800

 dense_2 (Dense)             (None, 256)               131328

 dense_3 (Dense)             (None, 11)                2827

=================================================================
Total params: 3805707 (14.52 MB)
Trainable params: 3805707 (14.52 MB)
Non-trainable params: 0 (0.00 Byte)
_________________________________________________________________
```

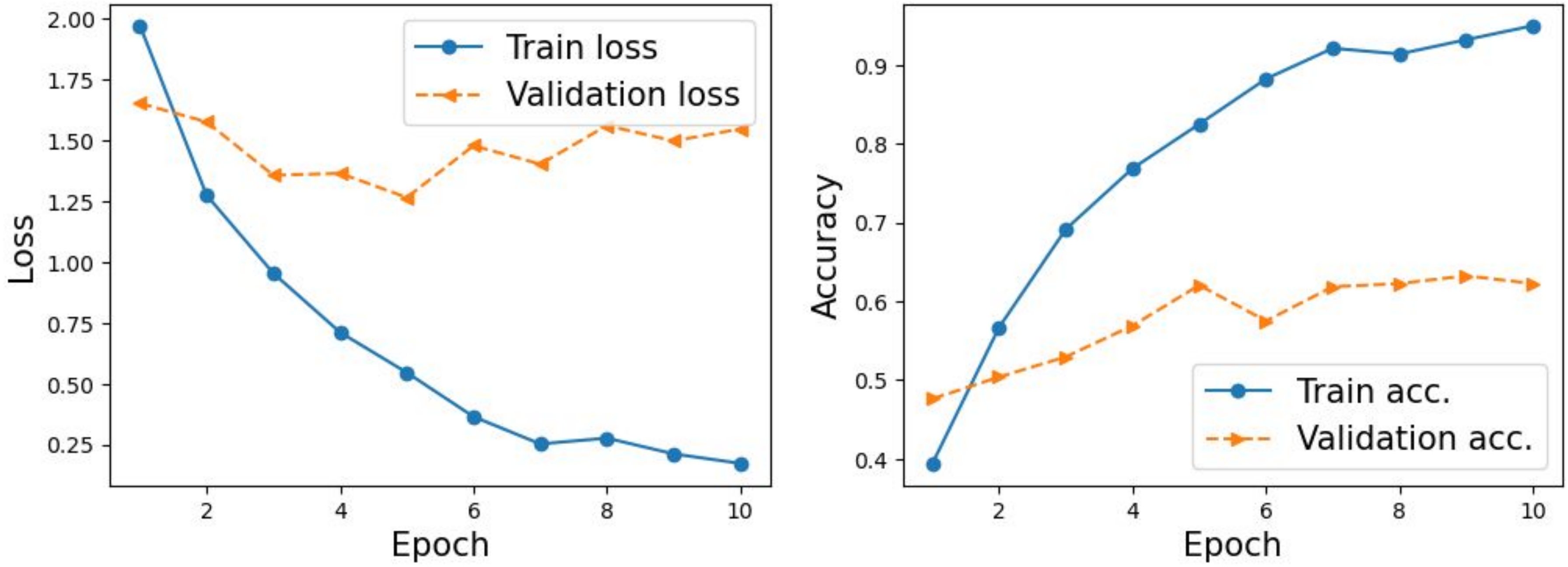



```
Test Accuracy: 0.4727
```

CNN

```
Model: "sequential_24"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv_1 (Conv2D)             (None, 100, 100, 32)      2432

 pool_1 (MaxPooling2D)       (None, 50, 50, 32)        0

 conv_2 (Conv2D)             (None, 50, 50, 64)        51264

 pool_2 (MaxPooling2D)       (None, 25, 25, 64)        0

 flatten_24 (Flatten)        (None, 40000)             0

 fc_1 (Dense)                (None, 1024)              40961024

 dropout_24 (Dropout)        (None, 1024)              0

 fc_2 (Dense)                (None, 11)                11275

=================================================================
Total params: 41,025,995
Trainable params: 41,025,995
Non-trainable params: 0
```

CNN



Berkeley
UNIVERSITY OF CALIFORNIA

Tuning and Improvements

Training Accuracy	Validation Accuracy	Img Dimension	# Per Sample	Kernel Size	Strides	Pool Size	Learning Rate	Optimizer	Brightness	Contrast Factor	Flip on Train	# of Params	Training Time
96.31%	64.82%	100x100	230	5,5	1,1	2,2	0.001	Adam	0.3	3	yes	41,025,995	7m
96.73%	51.01%	200x200	400	5,5	1,1	2,2	0.001	Adam	0.3	3	yes	163,905,955	69m
98.55%	65.61%	100x100	230	3,3	1,1	2,2	0.001	Adam	0.3	3	yes	40,991,691	5m
86.59%	65.61%	100x100	230	5,5	2,2	2,2	0.001	Adam	0.3	3	yes	2,425,291	37s
90.71%	70.55%	100x100	230	5,5	1,1	3,3	0.001	Adam	0.3	3	yes	7,995,851	3m
9.26%	8.30%	100x100	230	5,5	1,1	2,2	0.01	Adam	0.3	3	yes	41,025,995	7m
78.36%	63.24%	100x100	230	5,5	1,1	2,2	0.001	SGD	0.3	3	yes	41,025,995	7m
95.32%	61.66%	100x100	230	5,5	1,1	2,2	0.001	Adam	0.1	3	yes	41,025,995	7m
93.41%	57.51%	100x100	230	5,5	1,1	2,2	0.001	Adam	0.3	2	yes	41,025,995	7m
95.98%	60.08%	100x100	230	5,5	1,1	2,2	0.001	Adam	0.3	3	no	41,025,995	7m

Tuning and Improvements

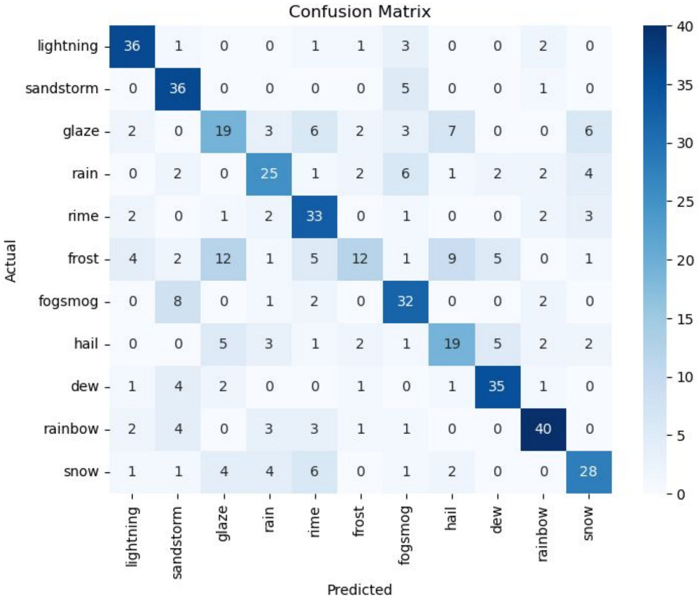
Training Accuracy	Validation Accuracy	Img Dimension	# Per Sample	Kernel Size	Strides	Pool Size	Learning Rate	Optimizer	Brightness	Contrast Factor	Flip on Train	# of Params	Training Time
90.71%	70.55%	100x100	230	5,5	1,1	3,3	0.001	Adam	0.3	3	yes	7,995,851	3m
96.08%	70.93%	100x100	230	3,3	1,1	3,3	0.001	Adam	0.3	3	yes	7,961,547	2m
80.47%	65.02%	100x100	230	5,5	2,2	3,3	0.001	Adam	0.3	3	yes	328,139	25s
9.40%	8.30%	100x100	230	5,5	1,1	3,3	0.01	Adam	0.3	3	yes	7,995,851	3m
69.14%	64.23%	100x100	230	5,5	1,1	3,3	0.001	SGD	0.3	3	yes	7,995,851	3m
91.30%	65.42%	100x100	230	5,5	1,1	3,3	0.001	Adam	0.1	3	yes	7,995,851	3m
92.75%	67.19%	100x100	230	5,5	1,1	3,3	0.001	Adam	0.3	2	yes	7,995,851	3m
92.85%	66.40%	100x100	230	5,5	1,1	3,3	0.001	Adam	0.3	3	no	7,995,851	3m

Tuning and Improvements

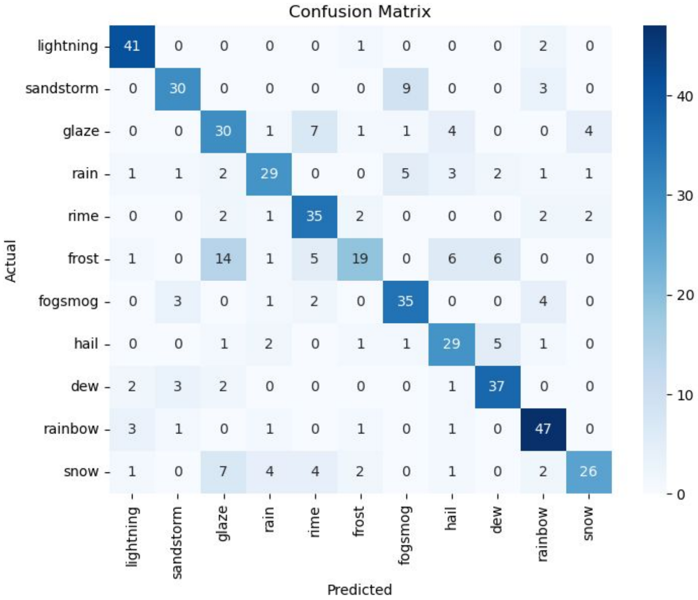
Training Accuracy	Validation Accuracy	Testing Accuracy	Img Dimension	# Per Sample	Kernel Size	Strides	Pool Size	Learning Rate	Optimizer	Brightness	Contrast Factor	Flip on Train	# of Params	Training Time
90.71%	70.55%	63.83%	100x100	230	5,5	1,1	3,3	0.001	Adam	0.3	3	yes	7,995,851	3m
96.08%	70.93%	69.96%	100x100	230	3,3	1,1	3,3	0.001	Adam	0.3	3	yes	7,961,547	2m

Berkeley
UNIVERSITY OF CALIFORNIA

Confusion Matrix: Before & After



Before



After

Precision, Recall, & F1-Scores

	Lightning	Sandstorm	Glaze	Rain	Rime	Frost	Fogsmog	Hail	Dew	Rainbow	Snow
Precision	0.84	0.79	0.52	0.72	0.66	0.7	0.69	0.64	0.74	0.76	0.79
Recall	0.93	0.71	0.62	0.64	0.8	0.37	0.78	0.72	0.82	0.87	0.55
F1-Score	0.88	0.75	0.57	0.68	0.72	0.48	0.73	0.68	0.78	0.81	0.65

Berkeley
UNIVERSITY OF CALIFORNIA

Conclusions

- Linear models could not handle image recognition as well as CNN
- Baseline models gave a **benchmark of 0.4 - 0.5 accuracy**
  - Picture resolutions did not make significant difference
  - Learning rate of 0.0001 noticeably improved performance stability
- The best CNN model after fine tuning could **improve accuracy to 0.7**
  - A kernel size of 3x3 and pool size of 3x3 capture the right balance of image details with reasonable computing time
- Future work
  - Further image augmentation (image cropping, further resizing, etc.)
  - Increase epoch and reduce learning rate further
  - Leverage additional convolutional layers (MeteCNN in paper used 13)
  - Utilize ensemble learning to see if accuracy increases significantly

Berkeley
UNIVERSITY OF CALIFORNIA

NeuroIPS checklist

- (a) Do the **main claims** made in the abstract and introduction accurately reflect the paper's contributions and scope?
  - Yes -claims in the paper match theoretical and experimental results in terms of how much the results can be expected to generalize.
  - The paper's purpose is started at the beginning of the presentation slides
- (b) Have you read the **ethics review guidelines** and ensured that your paper conforms to them?
  - Yes
- (c) Did you discuss any potential **negative societal impacts** of your work?
  - As far as we concern, our research piece was on weather images classification and it should have little relationship with social impacts that have potentially negative
- (d) Did you describe the **limitations** of your work?
  - No

If you are including theoretical results...

  - N/A

Berkeley
UNIVERSITY OF CALIFORNIA

NeuroIPS Code of Ethics - checklist (conti.)

If you ran experiments...

- (a) Did you include the code, data, and instructions needed to **reproduce** the main experimental results (either in the supplemental material or as a URL)?
  - The research design was original and was not to reproduce certain experimental results
- (b) Did you specify all the **training details** (e.g., data splits, hyperparameters, how they were chosen)?
  - Yes
- (c) Did you report **error bars** (e.g., with respect to the random seed after running experiments multiple times)?
  - No
- (d) Did you include the amount of **compute** and the type of **resources** used (e.g., type of GPUs, internal cluster, or cloud provider)?
  - No

NeuroIPS Code of Ethics - checklist (conti.)

If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...

- (a) If your work uses existing assets, did you **cite** the creators?
  - Slides 4 has provided detailed information including the author of the data and a URL.
- (b) Did you mention the **license** of the assets?
  - The author has made the data available public on Kaggle for public use
- (c) Did you include any **new assets** either in the supplemental material or as a URL?
  - No
- (d) Did you discuss whether and how **consent** was obtained from people whose data you're using/curating?
  - The dataset is publically available for use
- (e) Did you discuss whether the data you are using/curating contains **personally identifiable information** or **offensive content**?
  - The data contains images with no identifiable persons shown that might violate privacy

NeuroIPS Code of Ethics - checklist (conti.)

If you used crowdsourcing or conducted research with human subjects...

- N/A. Our project uses images only and has no human subjects involved

References

Codes we referenced from (mentioned in guideline)

- Xiao, H., Zhang, F., Shen, Z., et al. (2021). Classification of Weather Phenomenon From Images by Using Deep Convolutional Neural Network. *Earth and Space Science*.

Berkeley
UNIVERSITY OF CALIFORNIA