TRASH MATERIAL CLASSIFICATION VIA DEEP LEARNING

Abdulellah Abualshour
Visual Computing Center

OUTLINE

- Motivation
- Problem Statement
- Related Work
- Datasets
- Methodology
- Experiments
- Results
- Conclusion
- Future Work

MOTIVATION

- **Waste management** is a very important task to preserve the environment. Environment and natural materials played an important function in the development of human societies and in history on the whole [1].

  Therefore, trash material classification helps a lot with achieving the goal of a better environment by applying deep learning methods to help us build useful applications dedicated for categorizing different types of materials given suitable datasets.

PROBLEM STATEMENT

In this project, I try to find the **best and most efficient approaches** to image classification suitable for trash material classification. My hope in this project is to implement and develop very accurate and efficient classifiers for the task of material classification.

I also aim to beat state-of-the-art results.

RELATED WORK

- G. Schwartz and K. Nishino. **Recognizing material properties from images**. *IEEE transactions on pattern analysis and machine intelligence*, 2018.
- C. Liu, L. Sharan, E. H. Adelson, and R. Rosenholtz. **Exploring features in a bayesian framework for material recognition**. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 239–246, 2010.
- M. Yang and G. Thung. **Classification of trash for recyclability status**. *Stanford University*, 2016.
- C. Bircanoglu, M. Atay, F. Beser, O. Genc, and M. A. Kizrak. **RecycleNet: Intelligent waste sorting using deep neural networks**. *Innovations in Intelligent Systems and Applications*, pages 1–7, 2018.

DATASETS

(a) TrashNet dataset

State-of-the-art: DenseNet121 (95%)
Reported in RecycleNet
2527 IMAGES

DEFEATED

(b) MINC-2500 dataset

State-of-the-art: Deep-TEN (81.3%)
Rutgers University CVPR Paper!
57500 IMAGES

DEFEATED

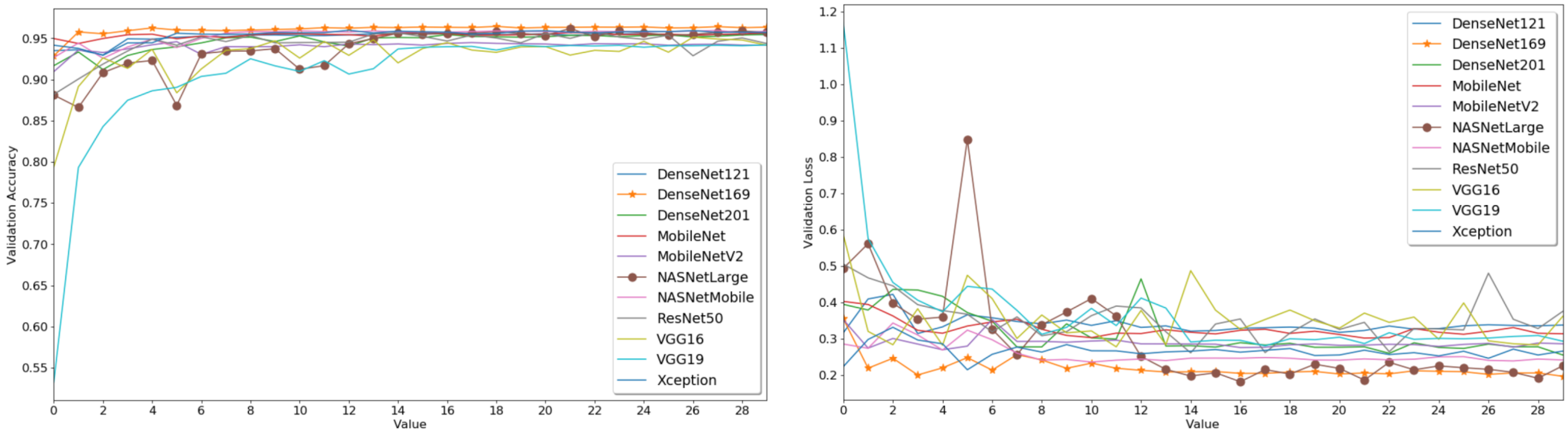
(c) FMD dataset

State-of-the-art: FV-CNN (82.4%)
1000 IMAGES

METHODOLOGY

- Convolutional Neural Networks
  - Keras
  - Fine-tuning
  - Lots of experimentation!
- Data Augmentation
  - Shifting images horizontally and vertically
  - Rotation by a 45 degree angle
  - Horizontal flipping
  - Shearing
  - Zooming
- Optimization
  - SGD, RMSProp, Adam, Adamax, Nadam

EXPERIMENTS: TRASHNET

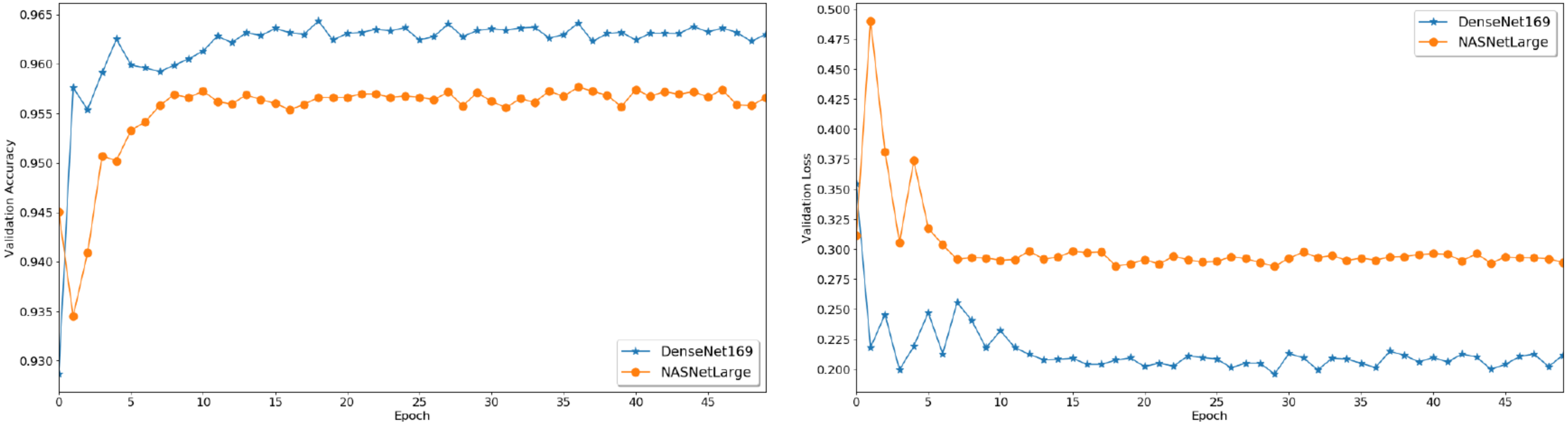


(a) Validation Accuracy

(b) Validation Loss

Figure 2. **Validation accuracy and loss during training using different fine-tuned architectures on the TrashNet [17] dataset.** NASNetLarge and DenseNet169 seem to perform similarly as the epoch number increases during training. When evaluating the models during the test phase, NASNetLarge perfroms the best on average.

EXPERIMENTS: TRASHNET (CONTD.)

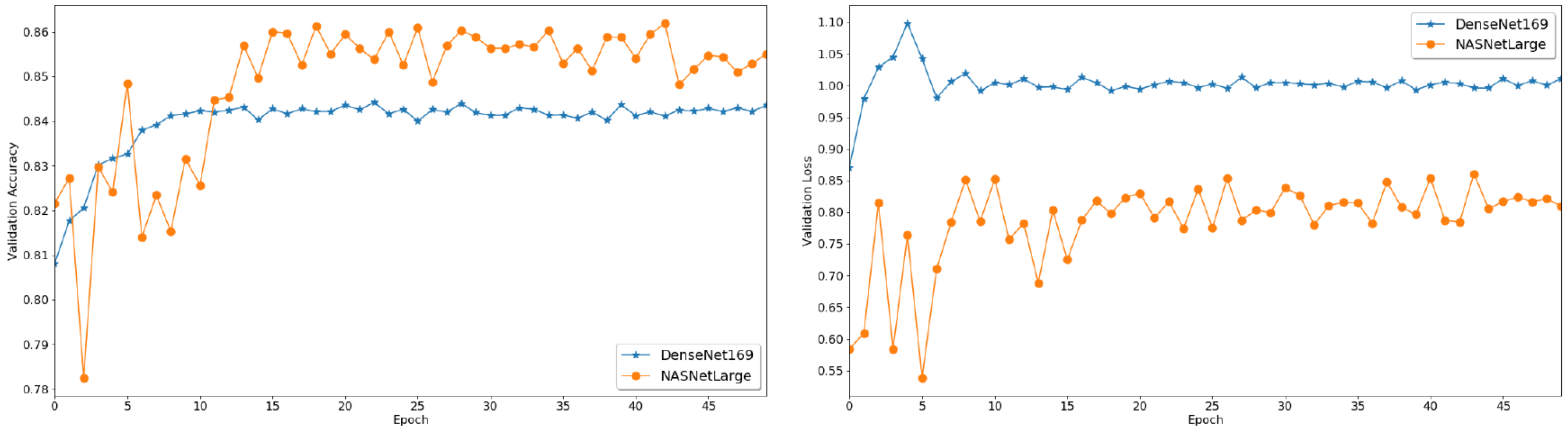


(a) Validation Accuracy

(b) Validation Loss

Figure 3. **Validation accuracy and loss during training the state-of-the-art beating models on TrashNet [16] dataset.** Both DenseNet169 and NASNetLarge beat state-of-the-art accuracy.

EXPERIMENTS: MINC-2500



(a) Validation Accuracy

(b) Validation Loss

Figure 4. **Validation accuracy and loss during training the state-of-the-art beating models on MINC-2500 [2] dataset.** Both DenseNet169 and NASNetLarge beat state-of-the-art accuracy.

RESULTS

Architecture	$\delta \uparrow$	cce↓	mse↓	mae↓	msle↓	Parameters	Batch Size
ResNet50 [8]	0.880	1.088	0.036	0.040	0.016	25m	32
VGG16 [16]	0.852	1.090	0.042	0.050	0.020	15m	32
VGG19	0.859	1.062	0.041	0.049	0.020	20m	32
MobileNet [9]	0.875	1.258	0.039	0.044	0.019	4m	32
MobileNetV2	0.866	1.062	0.039	0.046	0.019	3m	32
DenseNet121 [10]	0.870	1.146	0.041	0.045	0.020	7m	32
DenseNet169	0.885	1.123	0.035	0.039	0.017	13m	32
DenseNet201	0.881	0.945	0.035	0.040	0.017	18m	16
Xception [4]	0.880	0.806	0.033	0.042	0.016	23m	8
NASNetMobile [19]	0.877	0.987	0.035	0.041	0.017	5m	32
NASNetLarge	**0.901**	**0.791**	**0.031**	**0.038**	**0.015**	88m	4

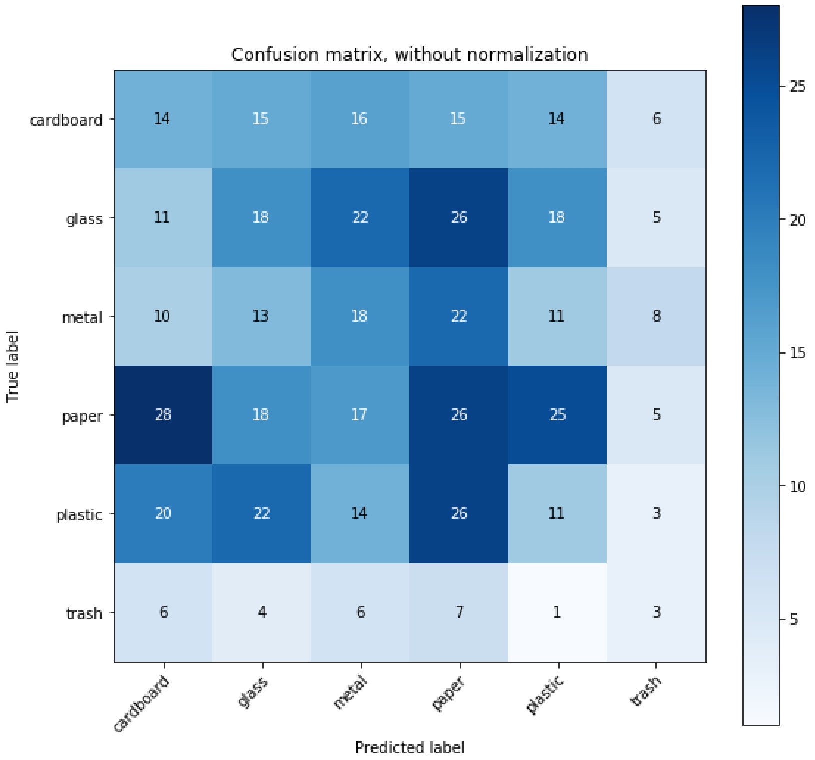
Table 1. **Comparisons of test results of different fine-tuned architectures on the TrashNet dataset.** Rows from left to right: architecture, test accuracy, categorical cross-entropy, mean squared error, mean absolute error, mean squared logarithmic error, number of trainable parameters in millions, and batch size. The reported numbers are real results from the experiments.

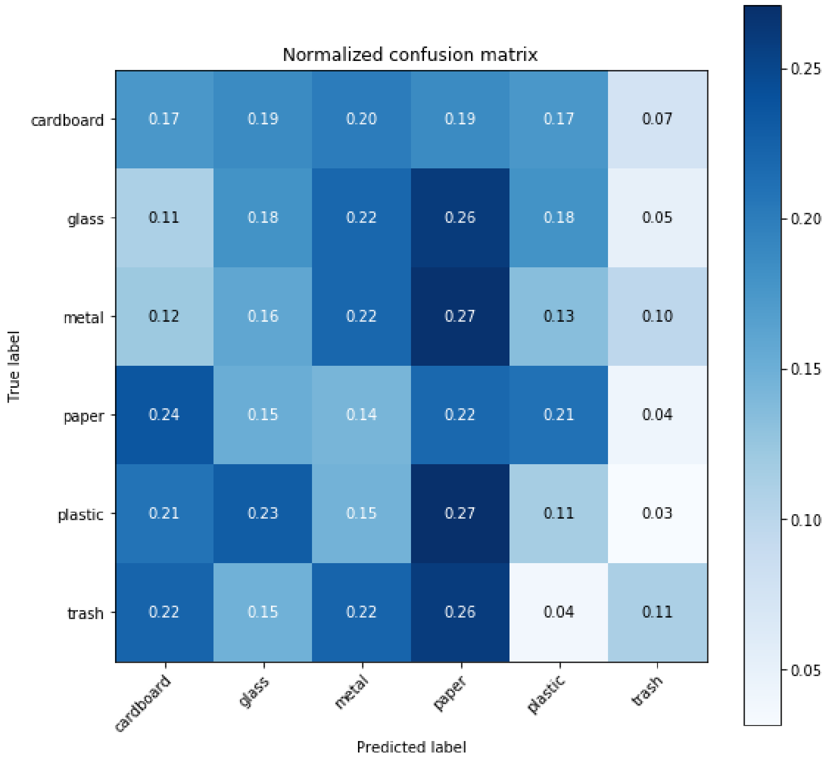
RESULTS (CONTD.)

Architecture	$\delta\uparrow$	cce↓	mse↓	mae↓	msle↓	Parameters	Batch Size
ResNet50 [8]	0.864	1.202	0.046	0.050	0.025	25m	32
VGG16 [15]	0.820	1.310	0.053	0.063	0.033	15m	32
VGG19	0.829	1.290	0.051	0.061	0.030	20m	32
MobileNet [9]	0.845	1.302	0.042	0.049	0.031	4m	32
MobileNetV2	0.844	1.112	0.043	0.049	0.027	3m	32
DenseNet121 [10]	0.850	1.226	0.052	0.055	0.032	7m	8
DenseNet169	0.873	1.243	0.046	0.040	0.020	13m	8
DenseNet201	0.870	1.145	0.040	0.038	0.018	18m	8
Xception [4]	0.866	0.906	0.035	0.049	0.022	23m	8
NASNetMobile [19]	0.852	1.023	0.040	0.043	0.019	5m	32
NASNetLarge	**0.891**	**0.802**	**0.034**	**0.038**	**0.017**	88m	4

Table 2. **Comparisons of test results of different fine-tuned architectures on the TrashNet dataset using SGD as an optimizer.** Rows from left to right: architecture, test accuracy, categorical cross-entropy, mean squared error, mean absolute error, mean squared logarithmic error, number of trainable parameters in millions, and batch size. The reported numbers are real results from the experiments.

RESULTS (CONTD.)





STATE-OF-THE-ART COMPARISONS

Architecture	$\delta \uparrow$
Zhang et al. (Deep-TEN) [18]	0.813
Ours (DenseNet169)	0.842
Ours (NASNetLarge)	**0.862**

Table 3. **Our results compared to state-of-the-art accuracy on MINC-2500 dataset.** We beat state-of-the-art.

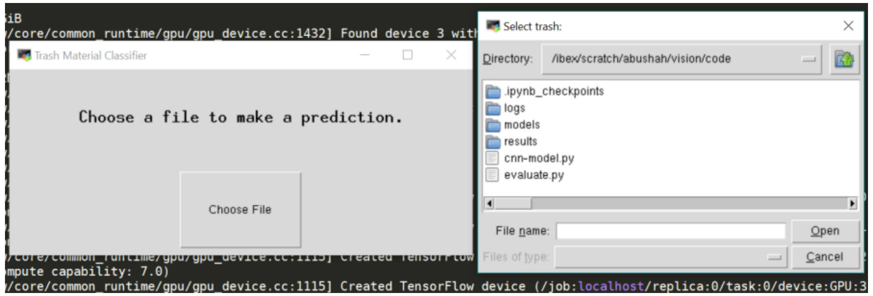
Architecture	$\delta \uparrow$
Bircanoglu et al. (DenseNet121) [3]	0.95
Ours (NASNetLarge)	0.955
Ours (DenseNet169)	**0.964**

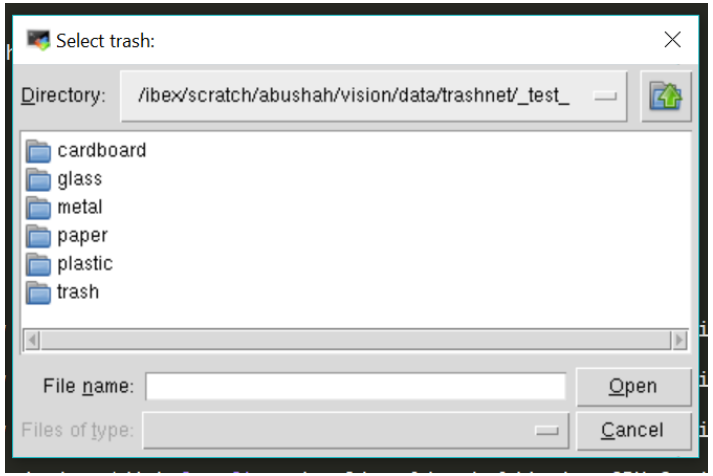
Table 4. **Our results compared to state-of-the-art accuracy on TrashNet dataset.** We beat state-of-the-art.

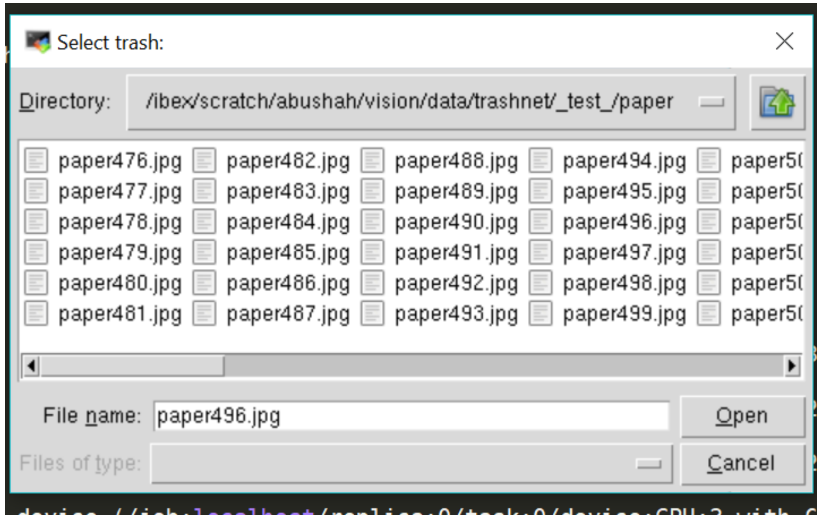
CONCLUSION

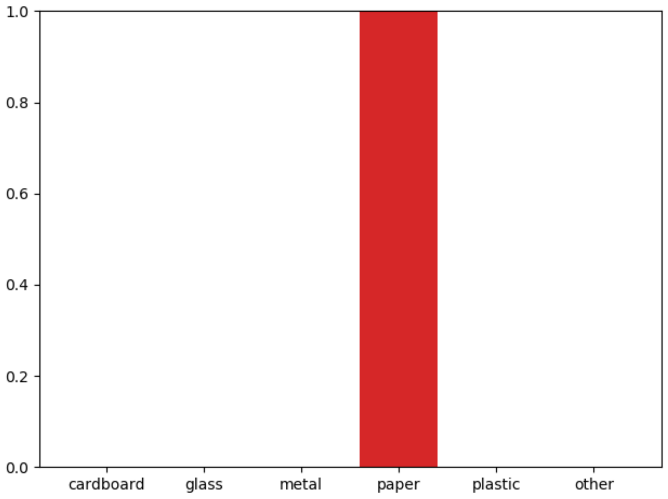
- Beat state-of-the-art results on two of the three benchmark datasets
  - (TrashNet and MINC-2500).
- Adam performed the best among all tested optimizing methods.
- NASNetLarge seemed to perform the best among all architectures used in MINC-2500.
- DenseNet169 on the other hand was better as an architecture for TrashNet.
- These better results were achieved by:
  - Providing better hardware for training, that resulted in my ability to increase batch size during training.
  - Elimination of some augmentations also helped increase the accuracy due to the nature of the datasets and how some augmentation techniques make the samples lose some features.

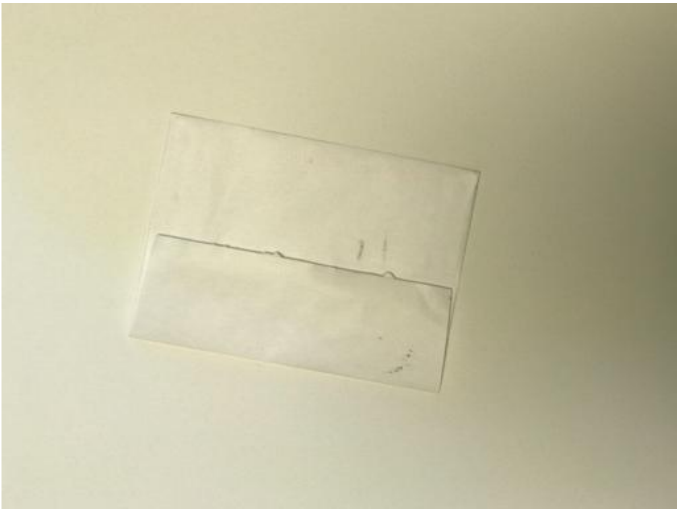
USER INTERFACE











REFERENCES

[1] S. Barles. *History of waste management and the social and cultural representations of waste*, volume 4. Springer, 2014.

[2] S. Bell, P. Upchurch, N. Snavely, and K. Bala. Material recognition in the wild with the materials in context database. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3479–3487, 2015.

[3] C. Bircanoglu, M. Atay, F. Beser, O. Genc, and M. A. Kizrak. Recyclenet: Intelligent waste sorting using deep neural networks. *2018 Innovations in Intelligent Systems and Applications (INISTA)*, pages 1–7, 2018.

[4] F. Chollet. Xception: Deep learning with depthwise separable convolutions. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1800–1807, 2017.

[5] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009.

[6] T. Dozat. Incorporating nesterov momentum into adam. In *ICLR*, 2015.

[7] A. Graves. Generating sequences with recurrent neural networks. *CoRR*, abs/1308.0850, 2013.

[8] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[9] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *CoRR*, abs/1704.04861, 2017.

[10] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger. Densely connected convolutional networks. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2261–2269, 2017.

[11] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2015.

[12] C. Liu, L. Sharan, E. H. Adelson, and R. Rosenholtz. Exploring features in a bayesian framework for material recognition. *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 239–246, 2010.

[13] H. Robbins and S. Monro. A stochastic approximation method. In *The Annals of Mathematical Statistics*, 2007.

[14] G. Schwartz and K. Nishino. Recognizing material properties from images. *IEEE transactions on pattern analysis and machine intelligence*, 2018.

[15] L. Sharan, R. Rosenholtz, and E. H. Adelson. Material perception: What can you see in a brief glance? *Journal of Vision*, vol. 14, no. 9, article 12, 2014.

[16] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2015.

[17] G. Thung. Trashnet. `https://github.com/garythung/trashnet`, 2016.

[18] M. Yang and G. Thung. Classification of trash for recyclability status. *Stanford University*, 2016.

[19] B. Zoph, V. Vasudevan, J. Shlens, and Q. V. Le. Learning transferable architectures for scalable image recognition. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8697–8710, 2018.