Report 01

Yiming Bian

April 22, 2022

Major Professor

Arun Somani(arun@iastate.edu)

Committee Members

Henry Duwe (duwe@iastate.edu)
Aditya Ramamoorthy (adityar@iastate.edu)
Alexander Stoytchev (alexs@iastate.edu)
Cindy Yu (cindyyu@iastate.edu)

1 For the past two weeks

Here are some experiments I have done in the past two weeks.

1.1 Re-trained models on other kind of noise

In Table 1 are all the data sets we tested so far. III and IV are greyed out because they are not the focus of this report. We also have 3 re-trained models: **RtO** (**R**e-**t**rained on I and II, which only contain **O**riginal images), **RtN** (**R**e-**t**rained on V and VI, which only contain salt&pepper, SNP **N**oise images) and **RtM** (**R**e-**t**rained on I, II, V and Vi, which only contain a **M**ixture of images).

Data set X is a new set that contains only images with Gaussian noise for 5 objects, each has 200 samples. Gaussian noise is different from SNP noise. Two parameters are required to generate Gaussian noise: mean and variance. Both parameters range from 0 to 1. Larger mean indicates more brightness and larger variance brings more noise. In Fig.1, it shows sixteen generated images with a level of Gaussian noise of a combination of mean and variance, whose values are $[0.2, 0.3, 0.4, 0.5]$.

As all the models have never seen Gaussian noise before, we are interested in their performance on data set X. Table 2 shows the prediction accuracy of three models on all noise data sets. There is a clear drop in accuracy on X, especially for RtO, which has never seen any noise data before.

Table 1: Set configurations: **Dis.** stands for distribution. Take 24×5 under **Objects Dis.** as an example, it stands for 5 objects and each has 24 samples. In **Noise Dis.** column, the tuple with five elements indicates the percentage of original image and image with each SNP noise level in the data set. For example, $(0, 25, 25, 25, 25)$ stands for the percentages of original image and images of each noise level from 1 to 4 are 0% and 25%.

Data set	**Purpose**	**# Samples**	**Objects Dis.**	**Noise Dis.**
I	train	500	100×5	$(100, 0, 0, 0, 0)$
II	validate	250	50×5	$(100, 0, 0, 0, 0)$
III	test	120	24×5	$(100, 0, 0, 0, 0)$
IV	test	$6,422$	$1300 \times 4 + 1222$	$(100, 0, 0, 0, 0)$
V	train	500	100×5	$(0, 25, 25, 25, 25)$
VI	validate	250	50×5	$(0, 25, 25, 25, 25)$
VII	test	298	$62 \times 4 + 50$	$(0, 24.16, 24.16, 25.84, 25.84)$
VIII	test	$17,122$	$3,466 \times 4 + 3,258$	$(0, 25.01, 25.01, 24.99, 24.99)$
IX	test	$8,566$	$1,734 \times 4 + 1,630$	$(0, 24.97, 24.97, 25.03, 25.03)$
X	test	$1,000$	200×5	100% Gaussian noise

Figure 1: Add sixteen levels of Gaussian noise to an example image: ”m” and ”v” stand for mean and variance respectively.

An additional experiment was on the normalization. An example on image normalization is shown in Fig.2. It has a very positive impact on test accuracy according to the experiment as shown in Table 2. There is a roughly 7% to 10% improvement.

Figure 2: An image with and without normalization

Table 2: Test accuracy of re-trained models

Data set	**RtO_acc**	**RtN_acc**	**RtM_acc**
VII	74.83%	88.59%	90.60%
VIII	76.19%	87.94%	90.43%
IX	76.83%	88.86%	90.89%
X	45.1%	75.2%	76.0%
X(**Normalized**)	52.1%	84.6%	85.9%

1.2 Accuracy on each level of noise

The aforementioned noise data sets, VII, VIII, IX and X, are mixtures of different levels of noise. We are interested in the performance of re-trained models in a finer granularity. Therefore, we construct new data sets with the same noise level, thus 20 test sets including 4 SNP noise data sets and 16 Gaussian noise data sets. Each test set contains 300 sample images for each of the 5 objects. We record the accuracy of all three models on all of these test sets and the results are shown in Table 3, Table 4, Table 5 and Table 6.

2 Plan for the next two weeks

Main tasks, so far, include

- Write a survey paper on low-quality image processing topic
- Train models on various large, noise training sets

Table 3: Test accuracy of re-trained models on SNP noise: The accuracy of RtO drop quickly as the noise level goes up while the other two models are more stable

SNP	**RtO**	**RtN**	**RtM**
0.1	89.67%	88.53%	93.13%
0.2	82.8%	89.07%	91.87%
0.3	75.93%	88.2%	89.87%
0.4	61.13%	87.93%	89.07%

Table 4: Test accuracy of RtO model on Gaussian noise: for RtO, which was trained only on original images, the lower brightness gives more accuracy, which is less intuitive

m \ v	0.2	0.3	0.4	0.5
0.2	69.27%	57.8%	49.40%	42.4%
0.3	66.38%	55.13%	45.67%	41.67%
0.4	64.44%	52.64%	47.33%	40.27%
0.4	59.44%	48.87%	43.73%	38.56%

Table 5: Test accuracy of RtN model on Gaussian noise: for RtN, which was trained only on SNP noise images, it has a very stable performance. However, it is still true that the lower brightness, the better performance.

m \ v	0.2	0.3	0.4	0.5
0.2	87.33%	83.67%	83.91%	82.6%
0.3	87.06%	84.27%	82%	82.73%
0.4	85.59%	84.12%	83%	81.73%
0.4	84.79%	83.73%	82.27%	81.32%

Table 6: Test accuracy of RtM model on Gaussian noise: for RtM, which was trained on a mixture of original and SNP noise data, it is still true about the brightness rule.

m \ v	0.2	0.3	0.4	0.5
0.2	89.07%	88%	86.11%	84.27%
0.3	88.33%	86.67%	84.47%	84.6%
0.4	86.26%	86.79%	84.8%	82.53%
0.4	84.86%	85.33%	84.33%	83.32%

- Experiment with more DNN architectures
- Look into Transformer

Writing a survey paper helps me to understand what has been done and what is going on in this area. Building a big picture of the area can not only prevent me from doing duplicate work, but also block some dead-end directions. It will be a descent achievement if it can be finished and submitted before the preliminary exam.

The current three models I have were trained on very small data sets to get preliminary results, and the results were exciting. Therefore, it is time to devote more to train the model. I have been arranging my code in the past weeks, it should be a quick process.

To consolidate my work, more DNN architectures should be tested to develop empirical conclusions, if not quantitative. Again, the code is ready and time is needed.

I have noticed people started to apply Transformer, which is commonly seen in natural language processing, to computer vision area. I have listed some significant papers in my reading list. They are [1], [2], [3], [4], [5], [6], [7], [8]. I will provide what I learn from these in the near future.

3 Resource and Rule

Github Repo: Project Repo
Report Frequency: Every two weeks
Next Report: May 6, 2022

References

[1] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, Advances in neural information processing systems 30 (2017).

[2] R. Bommasani, D. A. Hudson, E. Adeli, R. Altman, S. Arora, S. von Arx, M. S. Bernstein, J. Bohg, A. Bosselut, E. Brunskill, et al., On the opportunities and risks of foundation models, arXiv preprint arXiv:2108.07258 (2021).

[3] K. Han, A. Xiao, E. Wu, J. Guo, C. Xu, Y. Wang, Transformer in transformer, Advances in Neural Information Processing Systems 34 (2021).

[4] N. Parmar, A. Vaswani, J. Uszkoreit, L. Kaiser, N. Shazeer, A. Ku, D. Tran, Image transformer, in: International Conference on Machine Learning, PMLR, 2018, pp. 4055–4064.

[5] N. Kitaev, Ł. Kaiser, A. Levskaya, Reformer: The efficient transformer, arXiv preprint arXiv:2001.04451 (2020).

[6] K. Han, Y. Wang, H. Chen, X. Chen, J. Guo, Z. Liu, Y. Tang, A. Xiao, C. Xu, Y. Xu, et al., A survey on visual transformer, arXiv e-prints (2020) arXiv–2012.

[7] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, arXiv preprint arXiv:1810.04805 (2018).

[8] W. Fedus, B. Zoph, N. Shazeer, Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity, arXiv preprint arXiv:2101.03961 (2021).