COMP 546 HW 4

Kevin McCoy [kmm12]

colab notebook link

I did not collaborate with anyone.

1.0 PyTorch

1.1 Basics of Autograd

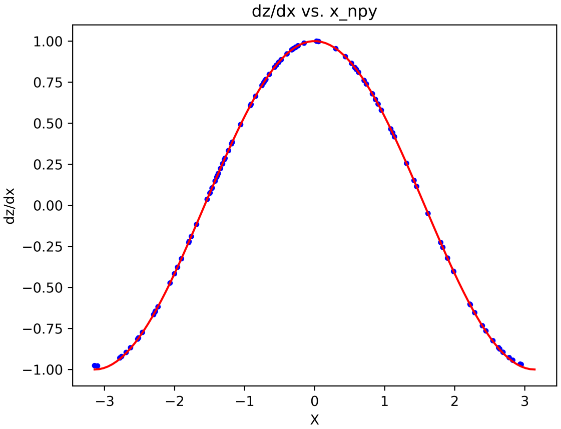

Figure 1: $\frac{dz}{dx}$ gradient overlaid on true plot of *cosine* function.

1.2 Image Denoising

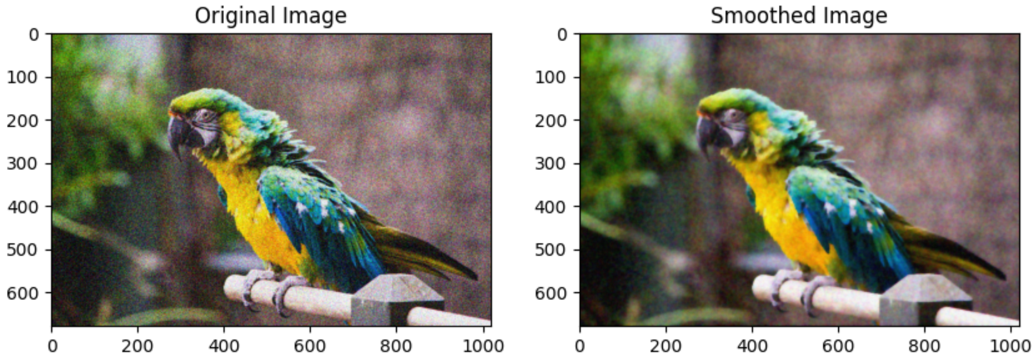

Figure 2: Smoothed Parrot Using $L1$ Loss, learning rate of 1 and α value of 1.

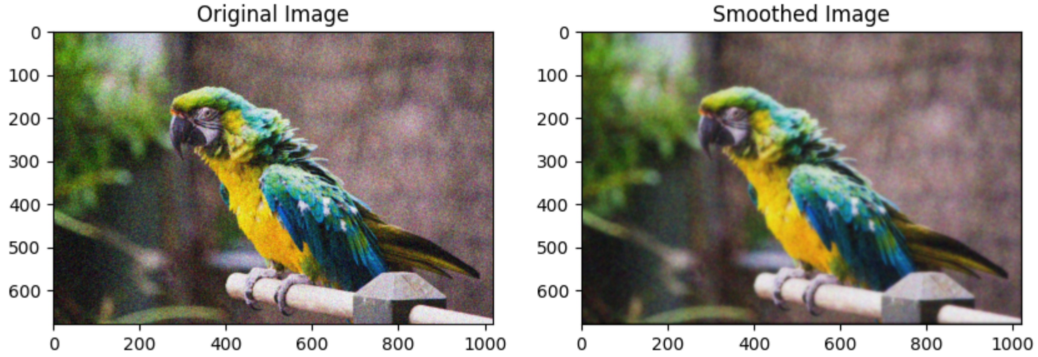

Figure 3: Smoothed Parrot Using $L2$ Loss, learning rate of 1 and α value of 1.

The L2 loss works worse than the L1 loss. This is because the L1 loss induces sparsity on the spatial gradients, which matches with the consistency assumption. The result is an image with less blur around edges.

2.0 Training an Image Classifier

a, b, c, d, e)

*** see notebook ***

f)

I will choose the last model because that has the least validation loss. If I had more computational resources, though, I would train for more epochs and choose the model with the lowest validation loss, even if it was not the latest model.

g)

	Predicted Class										
True Class	**airplane**	**automobile**	**bird**	**cat**	**deer**	**dog**	**frog**	**horse**	**ship**	**truck**	**class accuracy**
airplane	657	27	89	19	7	6	11	6	130	48	0.657
automobile	41	736	20	6	0	5	7	8	39	138	0.736
bird	62	13	581	71	59	85	47	37	27	18	0.581
cat	25	11	131	452	36	195	58	37	29	26	0.452
deer	29	10	205	78	410	57	70	116	18	7	0.410
dog	20	5	130	191	23	527	14	57	19	14	0.527
frog	6	8	95	104	44	31	668	12	11	21	0.668
horse	18	7	58	54	52	90	7	662	10	42	0.662
ship	74	45	18	12	3	8	5	4	789	42	0.789
truck	45	133	18	12	5	15	19	15	41	697	0.697

Table 1: Confusion Matrix with Class Accuracies, Overall accuracy: 0.618

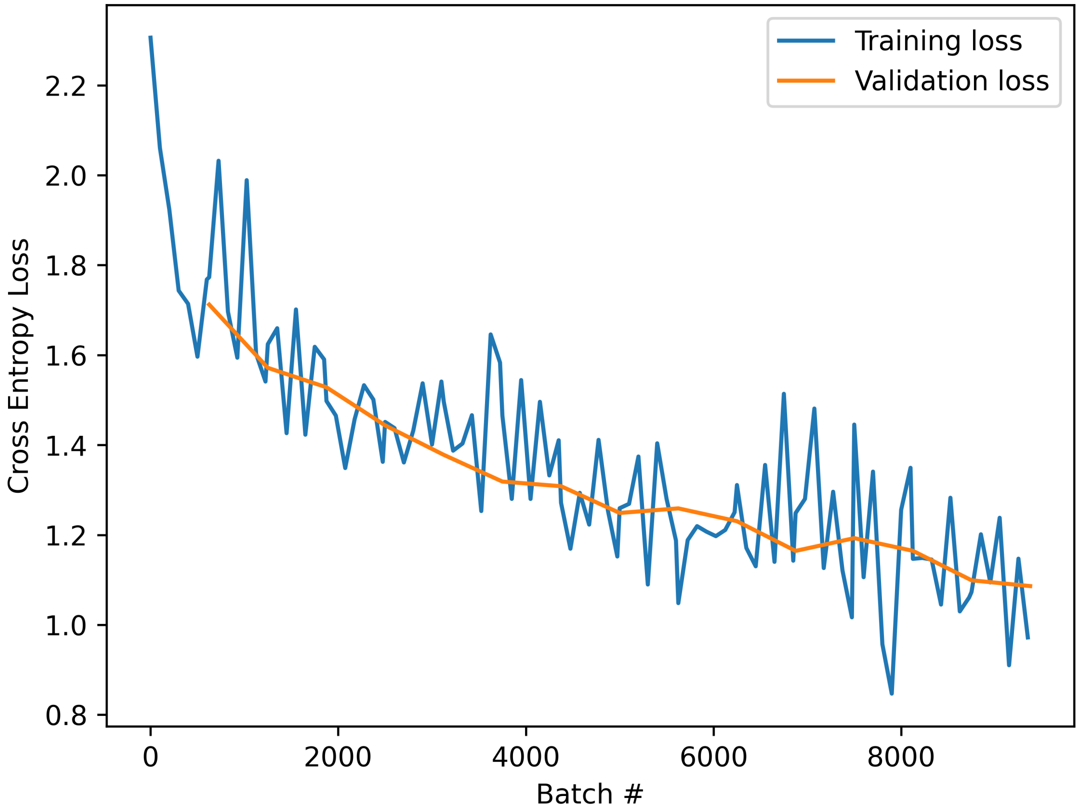

Figure 4: Training and validation loss of model.

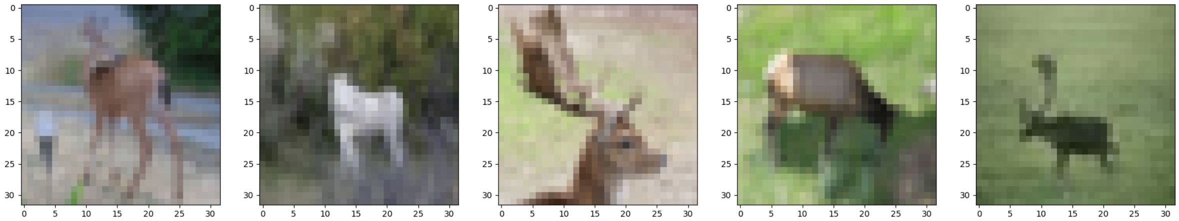

Figure 5: Pictures of deer confused as pictures of birds.

I believe the deer are confused as birds because of the background of the images. Both deer and birds are likely to be in images with a green background, as seen in the images above. Thus the model could have inadverdently learned the meaning of the background instead of the object in the foreground.

3.0 Adversarial Attacks

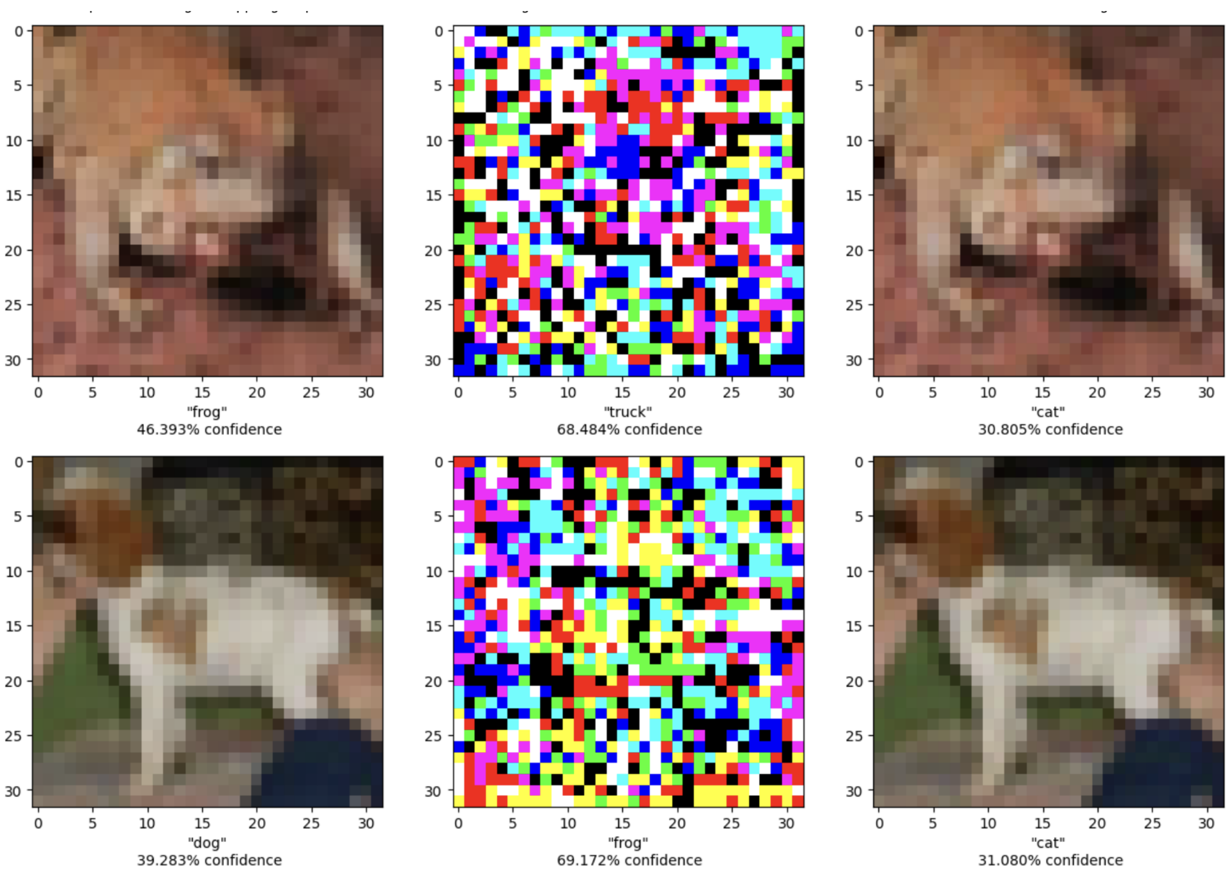

Figure 6: Examples of Adversarial Attacks. An ϵ value of 0.007 was used. The first column of images are the original images, gathered from the test set. The second column of images is the added noise term, before being multiplied by epsilon. The third column of images are the new images after adding the previous two images.