Motivation
With the increase in proliferation of fake images on the world wide web, there is a need to curb this spread more than ever. Fake and manipulated images are used to spread fake news. Most of these fake images are indistinguishable from real content to the unsuspecting reader. Manipulated images also aid document fraud. We therefore aim to develop a robust fake image detector that could potentially help reduce the spread of fake images and make people more informed.

Data
We have used the CASIA v2 dataset which has 7200 real images and 5531 altered (manipulated) images. Each altered image is either a single authentic image modified using professional image editing tools, or a combination of two authentic images.
We used a training/dev/test split of 80/10/10.

Preprocessing (ELA)
ELA (Error Level Analysis) is a technique that identifies areas within an image that are at different compression levels and can be applied to images with lossy compression. For a JPEG image, the entire picture should be at the same compression level. If a section of the image were to be at a significantly different level, it likely indicates a digital modification.

CNN Models
We tested across numerous CNN architectures. We started with creating a shallow CNN to gauge the performance of CNNs on fake image detection. After getting reasonable results, we shifted to implement transfer learning using numerous CNN architectures like Alexnet, Densenet, Resnet, Inceptionnet and VGG16.

Baseline Model:
We had implemented logistic regression using a simple neural network as our baseline model. This model was able to achieve an accuracy of 59% on the test data.

Resnet:
Deeper neural networks are known to be able to approximate functions better than shallower layers. However, with very deep networks, we stumble into the vanishing gradients problem where backpropagation makes the gradients for earlier layers infinitely small. Resnet50 was proposed as novel way to address this issue by using identity shortcut connections, parallel to the regular convolution layers, that can skip one or more layers. We used a 18-layer Resnet for our experiments.

Other CNN models that we used were Alexnet, VGG16, Inception and DenseNet.

Error Analysis
- Misclassified images were randomly distributed and did not belong to any particular type of image (e.g., nature, animal, etc)
- Most of the misclassified images were around 250x350 in resolution i.e., they had very low resolution as compared to the other images - 800x700 px
- In some images, ELA was unable to detect manipulations and thus led to a misclassification.

Results and Analysis
Our best model was the Resnet model with the following hyperparameters:

- Learning rate = 0.00025, Mini-batch size = 100, L2 Regularization = 0.005
- It had the following metrics after training:
  - Training Accuracy: 98%
  - Dev Set Accuracy: 96%
  - Test Accuracy: 89%
  - AUC: 0.9398
  - F1 Score: 0.9283
  - Recall: 0.9335

Future Work
- Trying ensemble techniques
  Technologies like Bagging and Boosting have shown great promise in tackling variance issues in models and generalizing well to unseen data. Therefore, creating numerous models and leveraging ensemble techniques can further help to develop a more robust model and also increase the classification accuracy.
- Using more sophisticated preprocessing techniques
  Our best model relied on hand engineered features. An area for future research is training an end-to-end model that performs equally as well on this task. This would to other image types would error level analysis is not as effective.