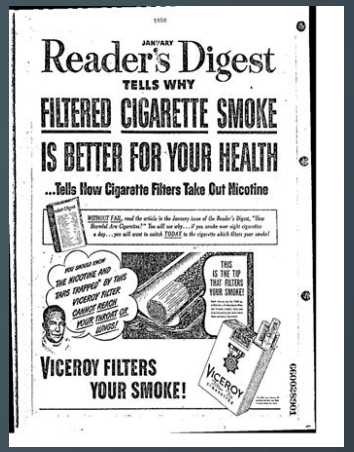
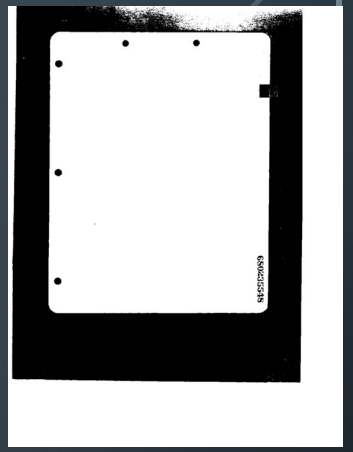
# DOCUMENT IDENTIFICATION WITH CONVOLUTIONAL NEURAL NETWORKS

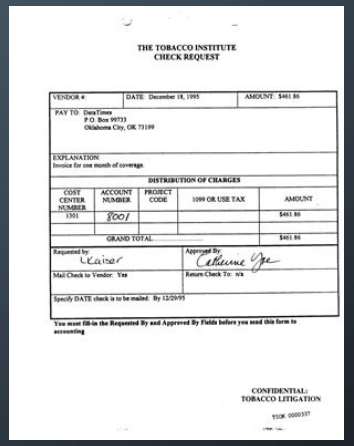
Group-5
Renu Gopal Reddy Durgampudi
Jack Curm
Geetha Ganji

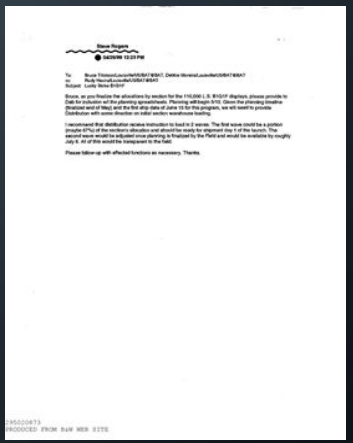
# RVL-CDIP DATASET

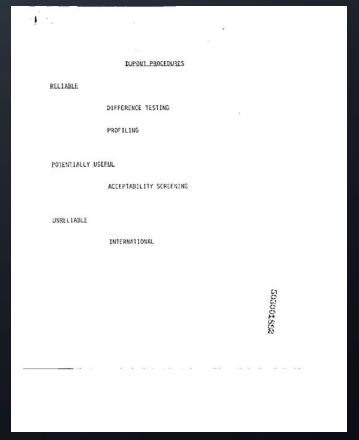
- Ryerson Vision Lab Complex Document Information Processing
- Data consists of digitally scanned documents of various types.
- Presplit into 320,000 training, 40,000 validation, and 40,000 testing images
  - 20,000 training per class, 2500 testing and validation per class
- 16 document classes: letter, form, email, handwritten, advertisement, scientific report, scientific publication, specification, file folder, news article, budget, invoice, presentation, questionnaire, resume, and memo

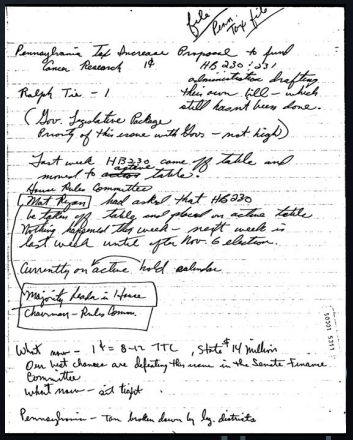






# DATA CLEANING

0. letter
1. form
2. email
3. handwritten
4. advertisement
5. scientific report
6. scientific publication
7. specification
8. file folder
9. news article
10. budget
11. invoice
12. presentation
13. questionnaire
14. resume
15. memo

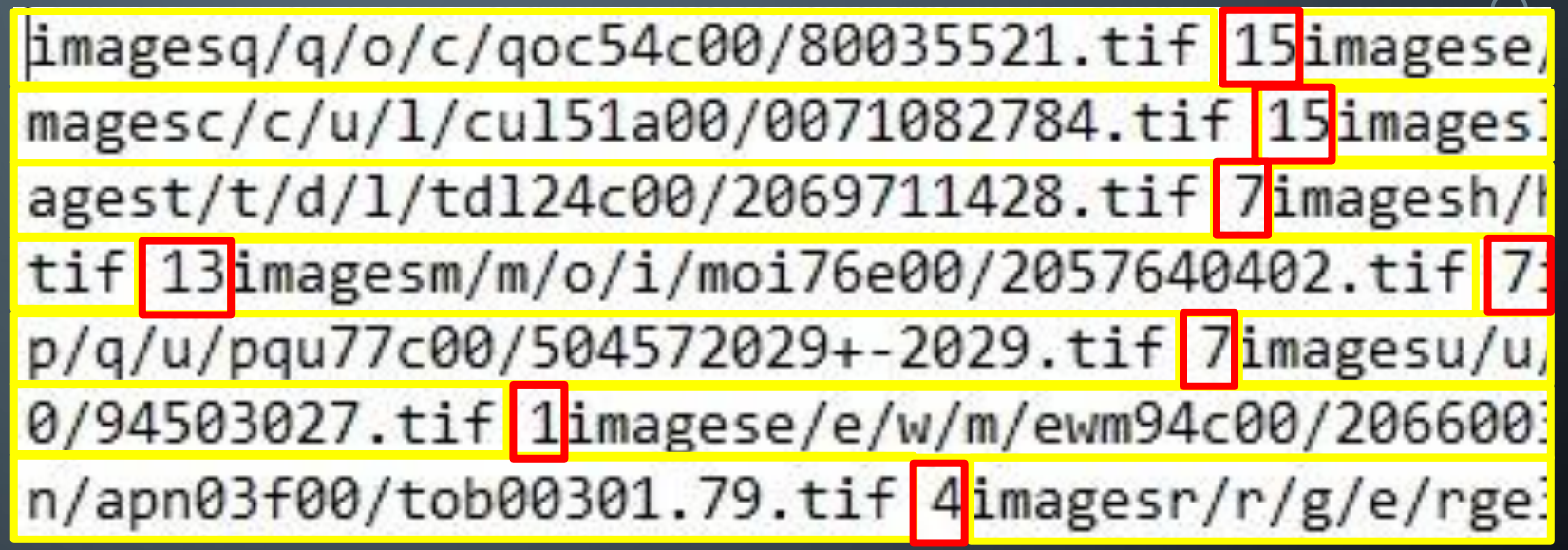
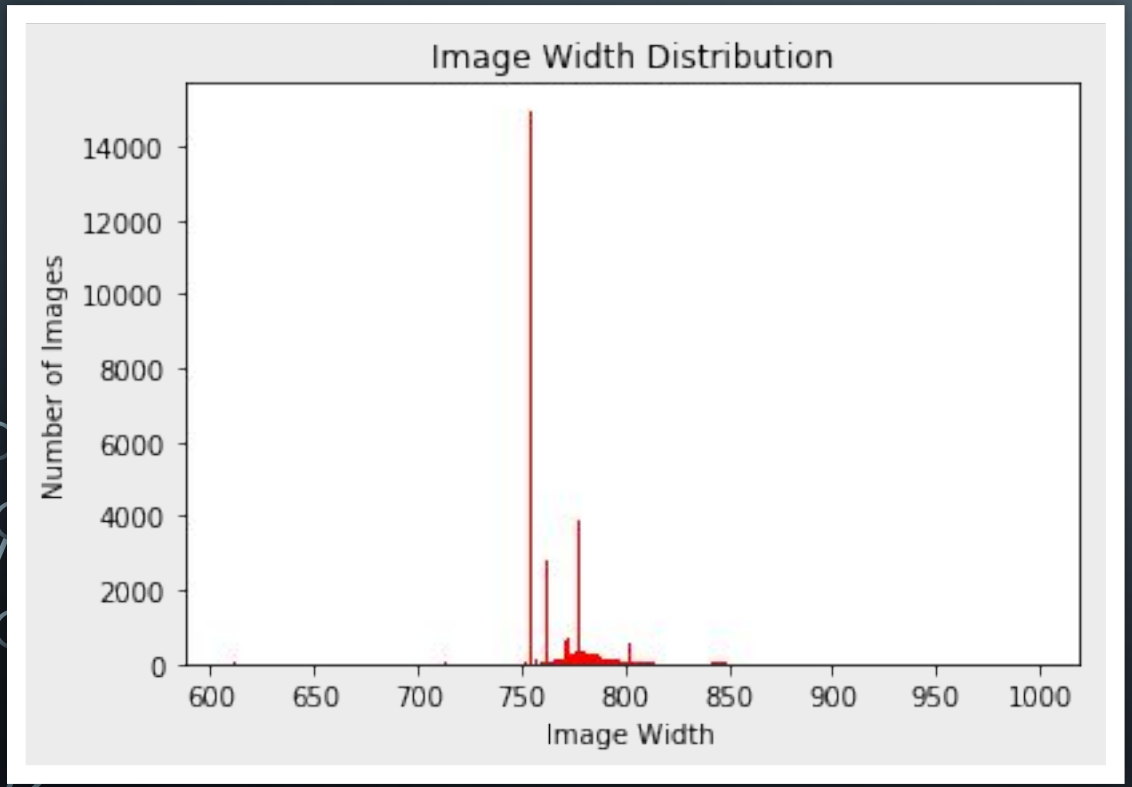
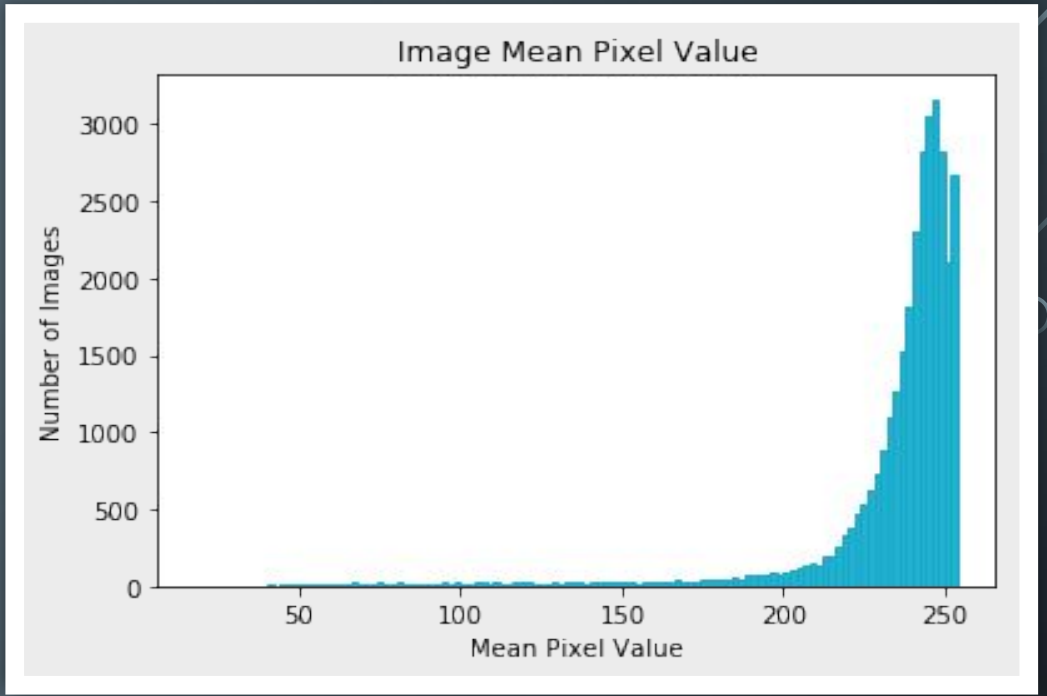
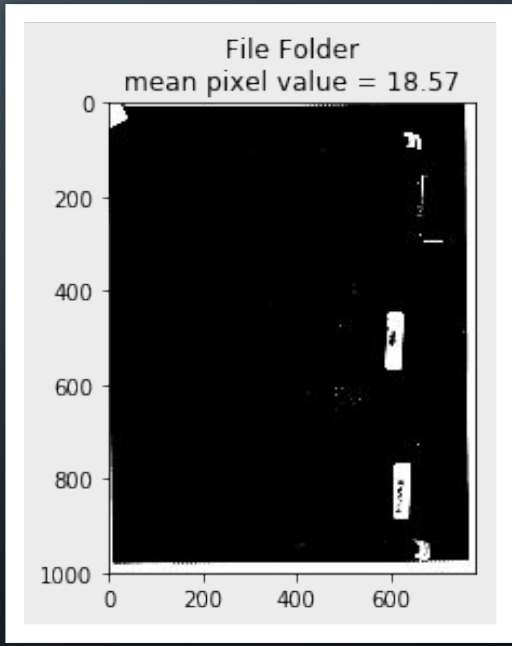


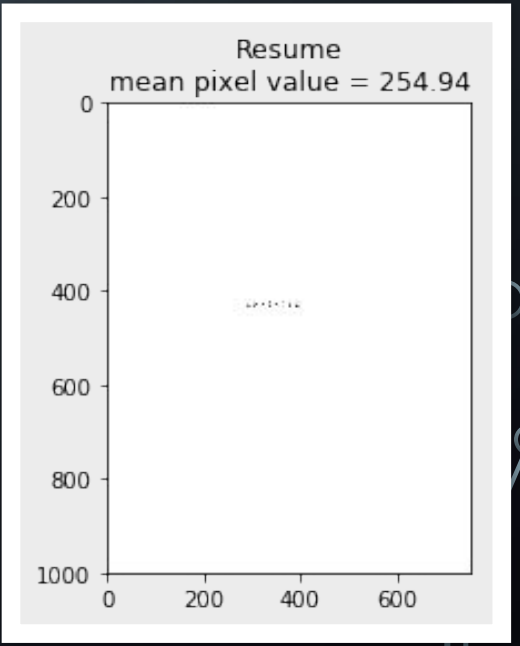

*Image paths in yellow, class labels in red*

- All images are 1000 pixels tall
- Most are 1000x752
- The majority of images are mostly  weight space
  - Darkest image (18.57 mean value)
  - Whitest image (254.94 mean values)
  - 0 is pure black, 255 is pure white on grayscale



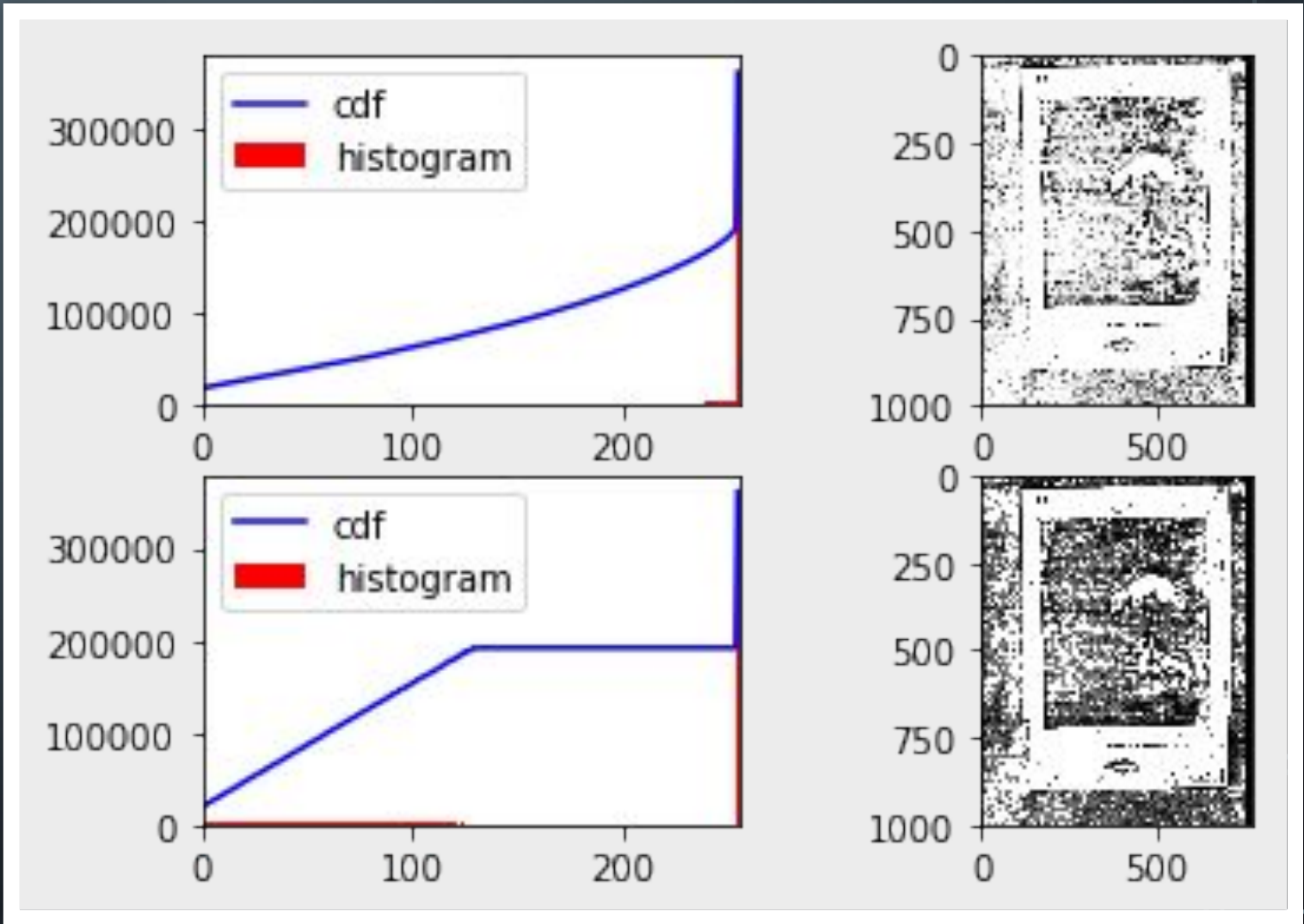



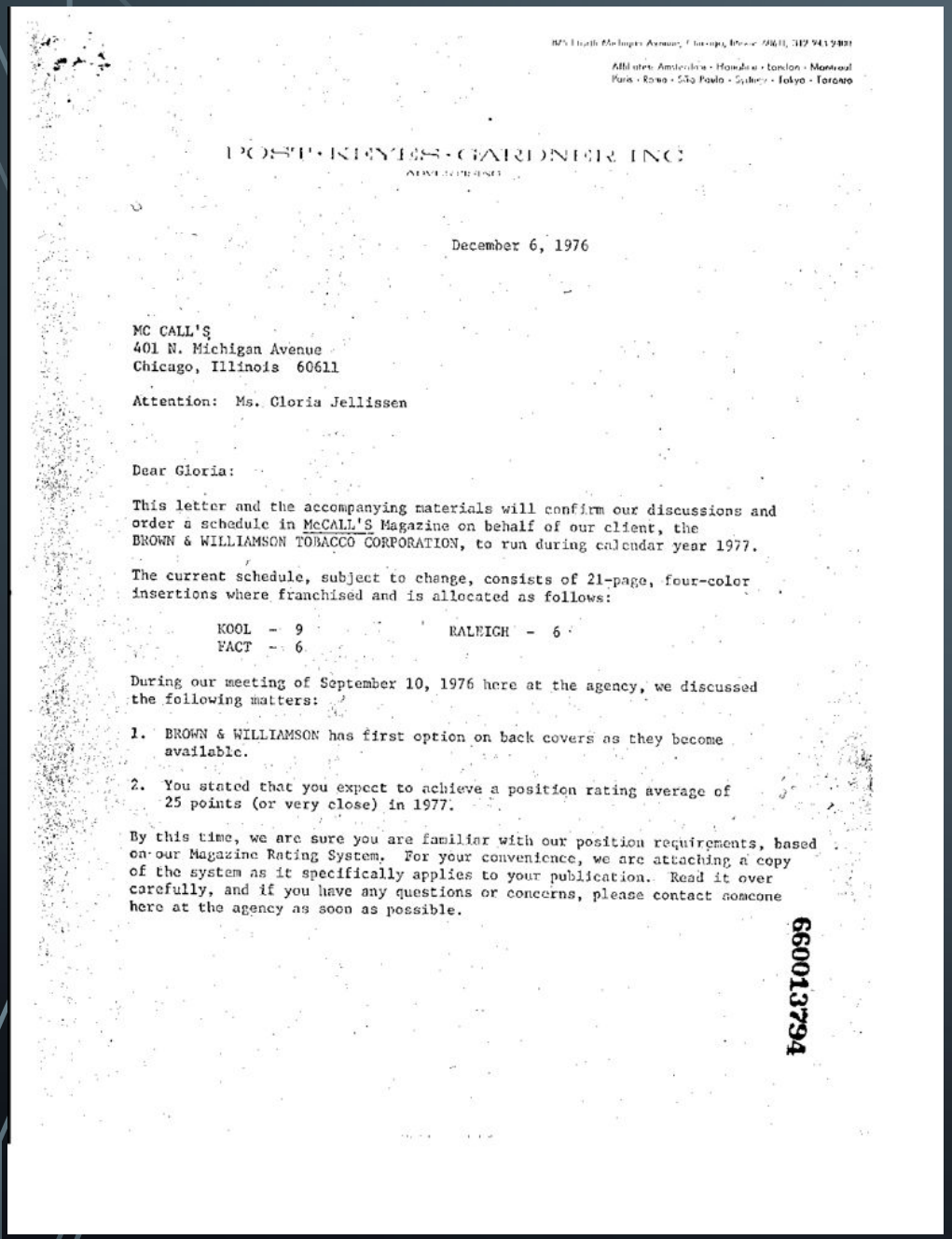



# PREPROCESSING

- Histogram Equalization
- Attempting to make the cumulative of pixel values as linear as possible to spread the distribution of influential features



(1000, 782)



POST·KEYES·GARDNER INC

December 6, 1976

MC CALL'S
401 N. Michigan Avenue
Chicago, Illinois 60611

Attention: Ms. Gloria Jellissen

Dear Gloria:

This letter and the accompanying materials will confirm our discussions and order a schedule in McCALL'S Magazine on behalf of our client, the BROWN & WILLIAMSON TOBACCO CORPORATION, to run during calendar year 1977.

The current schedule, subject to change, consists of 21-page, four-color insertions where franchised and is allocated as follows:

KOOL - 9 RALEIGH - 6
FACT - 6

During our meeting of September 10, 1976 here at the agency, we discussed the following matters:

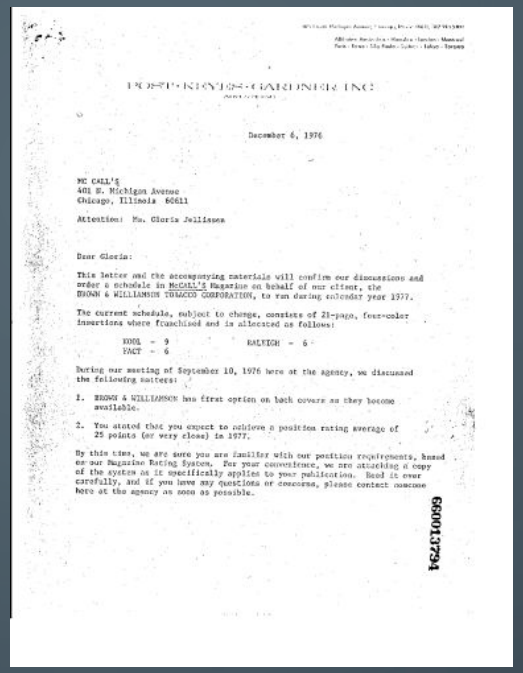
1. BROWN & WILLIAMSON has first option on back covers as they become available.

2. You stated that you expect to achieve a position rating average of 25 points (or very close) in 1977.

By this time, we are sure you are familiar with our position requirements, based on our Magazine Rating System. For your convenience, we are attaching a copy of the system as it specifically applies to your publication. Read it over carefully, and if you have any questions or concerns, please contact someone here at the agency as soon as possible.

660013794

(500, 381)



POST·KEYES·GARDNER INC

December 6, 1976

MC CALL'S
401 N. Michigan Avenue
Chicago, Illinois 60611

Attention: Ms. Gloria Jellissen

Dear Gloria:

This letter and the accompanying materials will confirm our discussions and order a schedule in McCALL'S Magazine on behalf of our client, the BROWN & WILLIAMSON TOBACCO CORPORATION, to run during calendar year 1977.

The current schedule, subject to change, consists of 21-page, four-color insertions where franchised and is allocated as follows:

KOOL - 9 RALEIGH - 6
FACT - 6

During our meeting of September 10, 1976 here at the agency, we discussed the following matters:

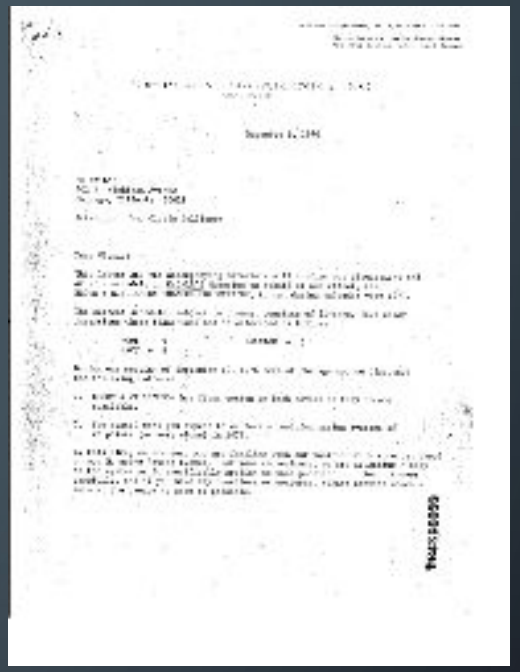
1. BROWN & WILLIAMSON has first option on back covers as they become available.

2. You stated that you expect to achieve a position rating average of 25 points (or very close) in 1977.

By this time, we are sure you are familiar with our position requirements, based on our Magazine Rating System. For your convenience, we are attaching a copy of the system as it specifically applies to your publication. Read it over carefully, and if you have any questions or concerns, please contact someone here at the agency as soon as possible.

660013794

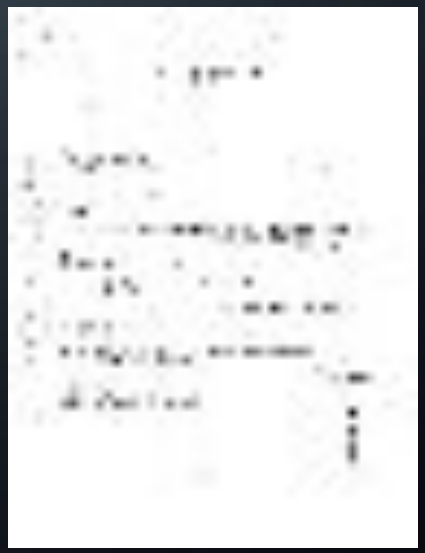
(250, 190)



660013794

(125, 95)

(62, 47)



# FINAL DATASET

- 32,000 training images (2000 per class)
- 12,000 testing images (750 per class)
- 12,000 validation (750 per class)
- All images resized to 224 x 224 for initial analysis

# PyTorch

- Loading custom dataset requires a custom data loader wrapped around PyTorch's Dataset class.

- Create .csv for each dataset with two columns:
  - Full image path
  - Class label

- Read and process with OpenCV

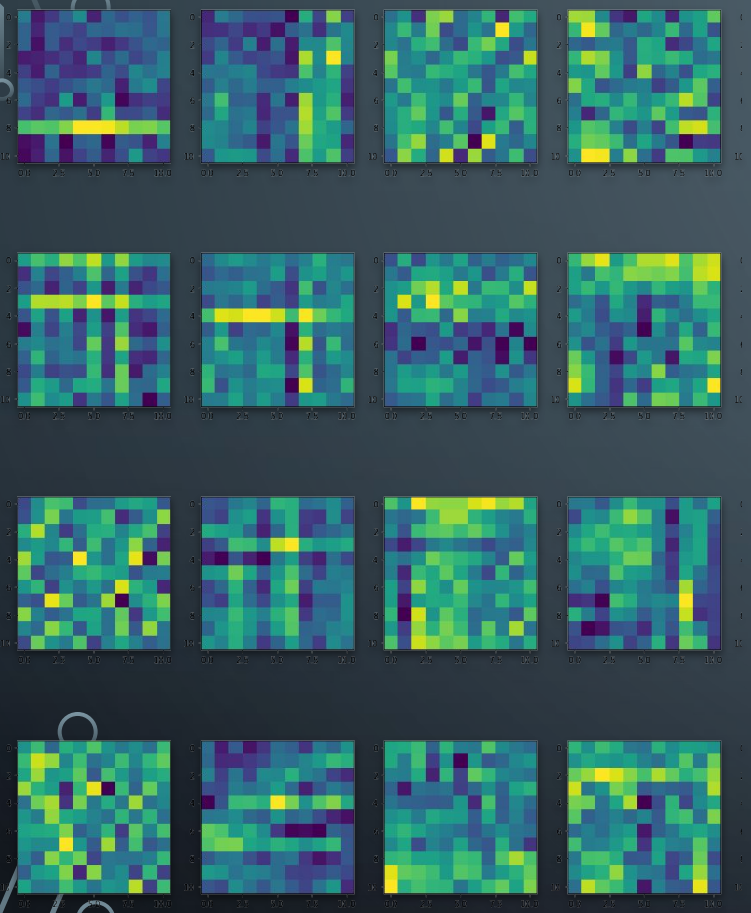
- OpenCV: *(height, width, # channels)*
- PyTorch: (*# channels, height, width)*
  - Need to reshape

```
class CustomDatasetFromImages(Dataset):
    def __init__(self, csv_path, transforms):
        # Read the csv file
        self.data_info = pd.read_csv(csv_path)
        # First column contains the image paths
        self.image_array = np.asarray(self.data_info.iloc[:, 0])
        # Second column is the labels
        self.label_array = np.asarray(self.data_info.iloc[:, 1])
        # Calculate len
        self.data_len = len(self.data_info.index)
        self.transforms = transforms

    def __getitem__(self, index):
        # Get image name from the pandas df
        single_image_name = self.image_array[index]
        img_as_img = cv2.imread(single_image_name, 0)
        img_resized = cv2.resize(img_as_img, (100, 100))
        img_final = np.expand_dims(img_resized, 0)
        single_image_label = self.label_array[index]
        # Return image and the label
        return (img_final, single_image_label)


    def __len__(self):
        return self.data_len

if __name__ == '__main__':
    train_loader = CustomDatasetFromImages('/home/ubuntu/Machine_Learning_
    test_loader = CustomDatasetFromImages('/home/ubuntu/Machine_Learning_I
```

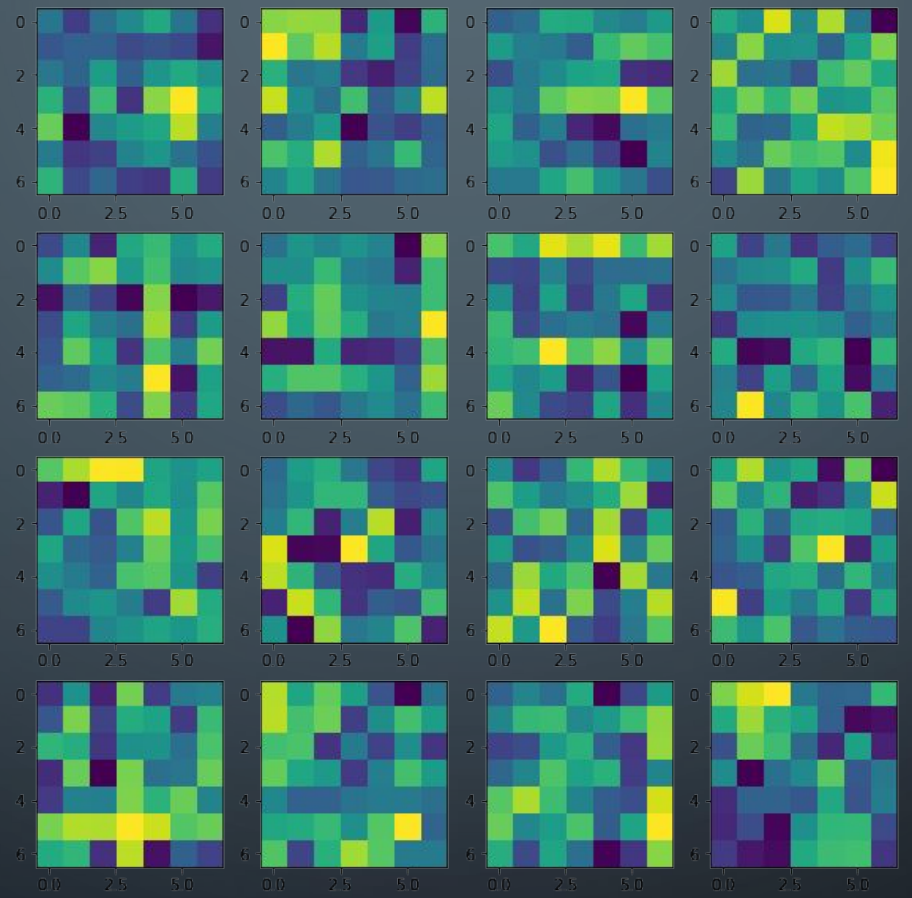
# Tuning Training Features

- Preprocessing
  - Histogram Equalization
- Image Size
  - 50, 100, 224
- Batch Size
  - 10, 50, 100
- Learning Rate
  - 1e-2, 1e-3, 1e-4, 1e-5
- Kernel Size
  - 11-9-7-3
  - 7-7-5-5-3
  - 3-3-3-3-3

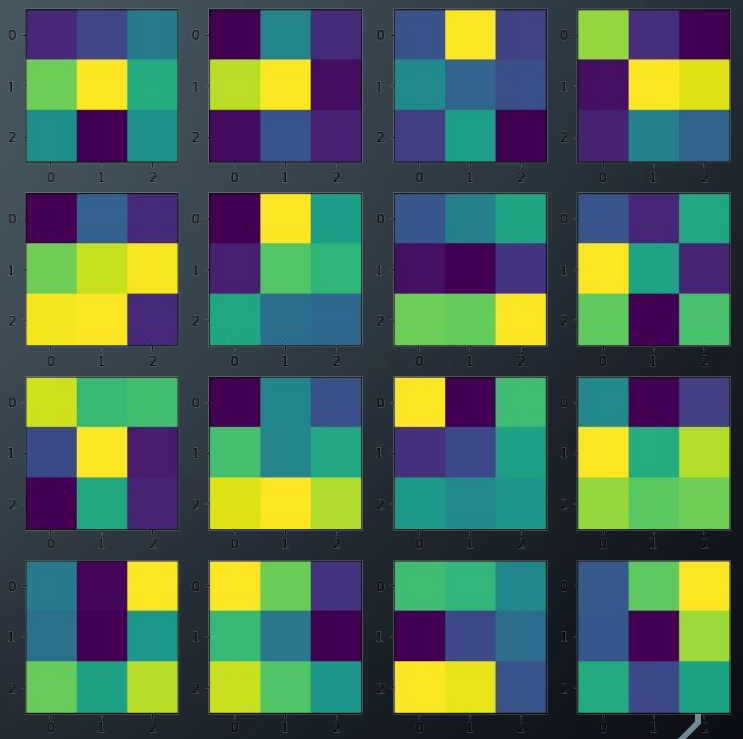
11 x 11 Kernel



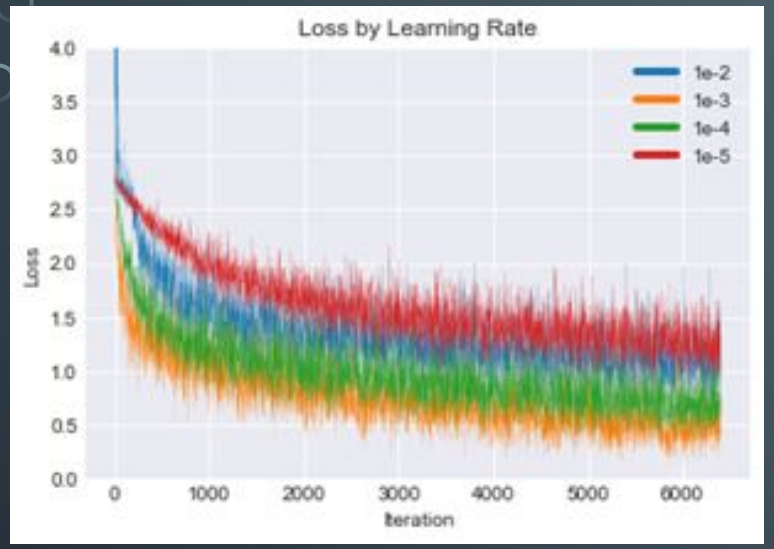
7 x 7 Kernel



3 x 3 Kernel

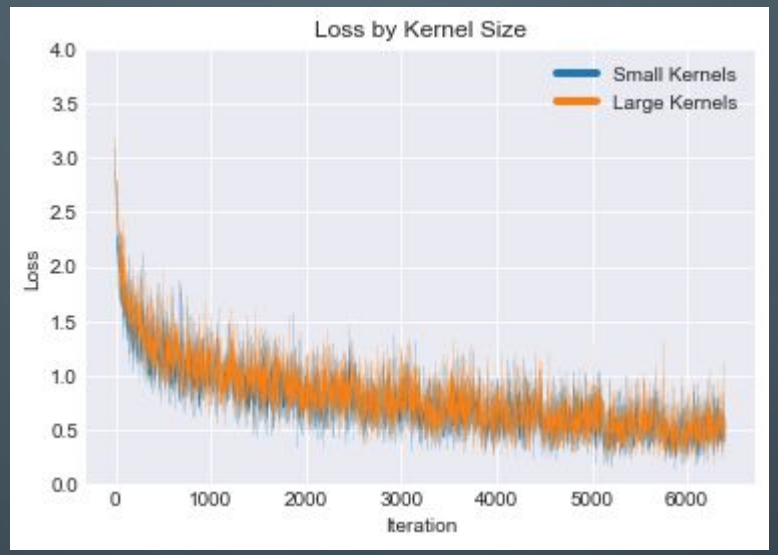


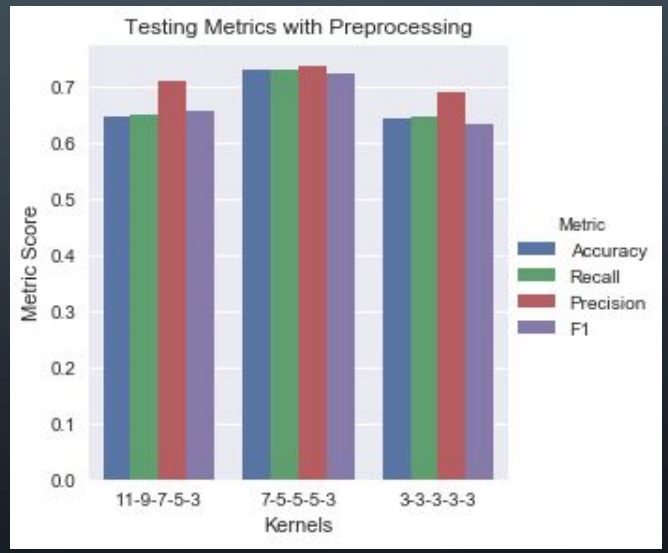
# Learning Rate



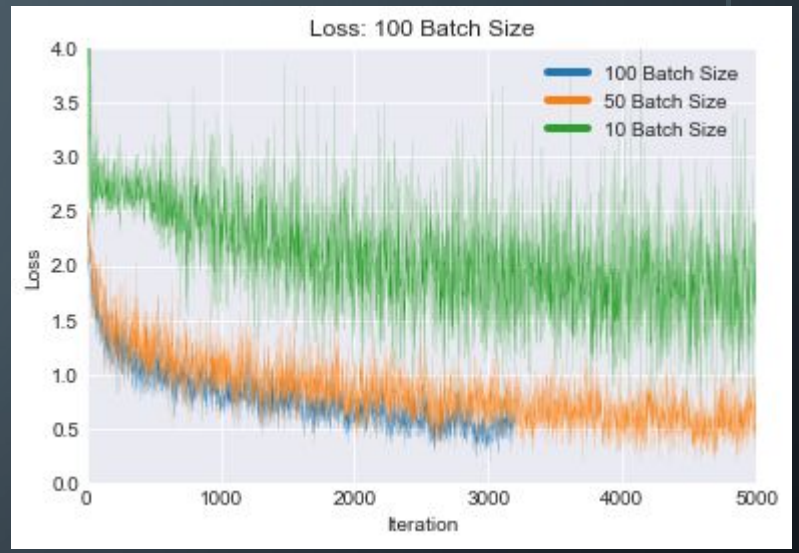


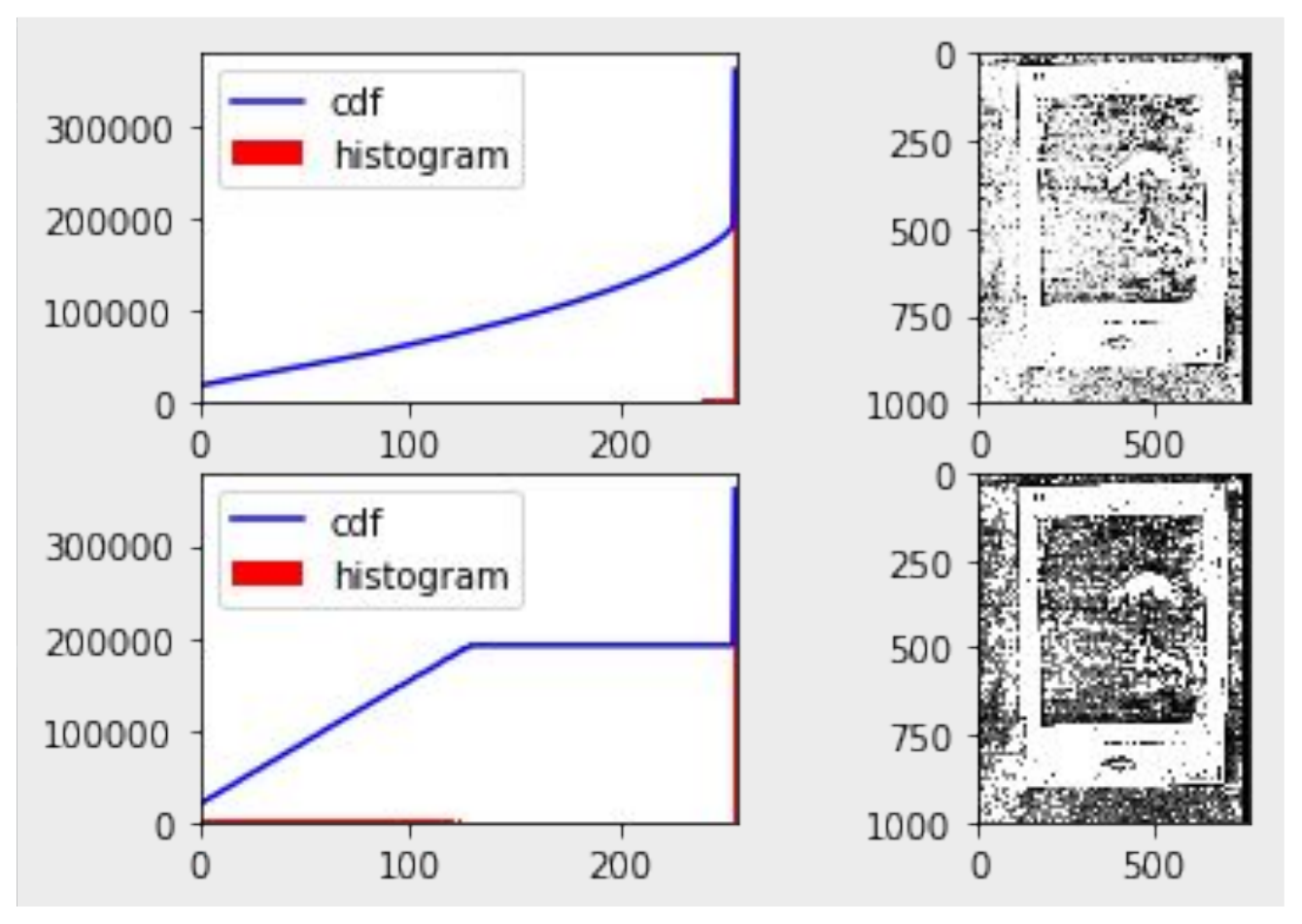
# Kernel Size

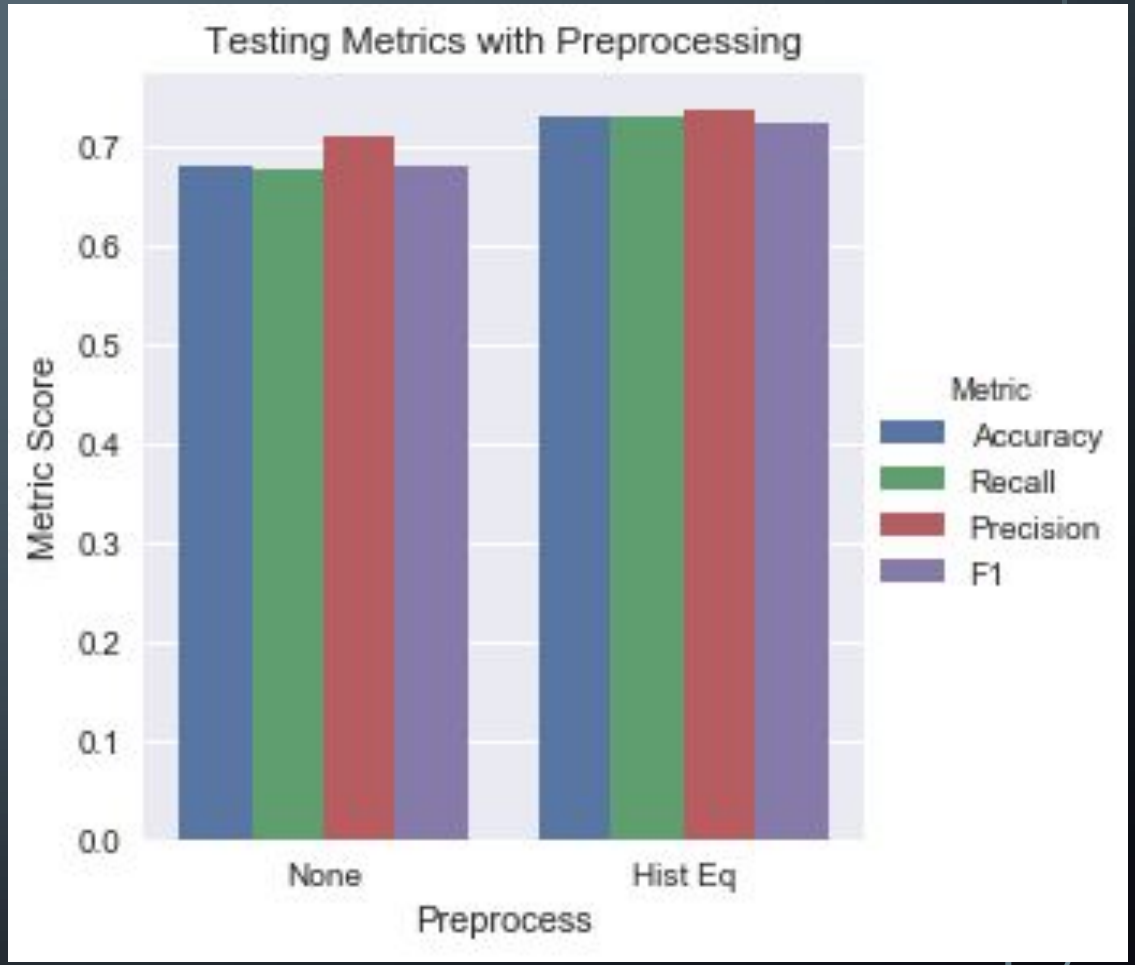



# Batch Size





cdf
histogram
300000
200000
100000
0
0
100
200
0
250
500
750
1000
0
500

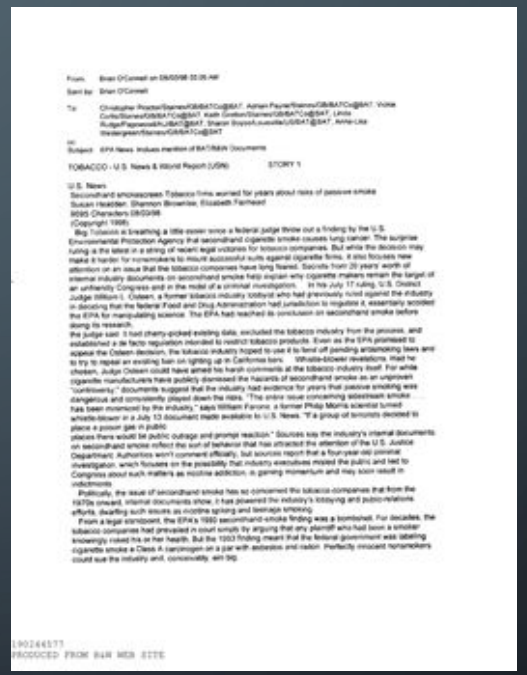

Testing Metrics with Preprocessing
Metric Score
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
None
Hist Eq
Preprocess
Metric
Accuracy
Recall
Precision
F1

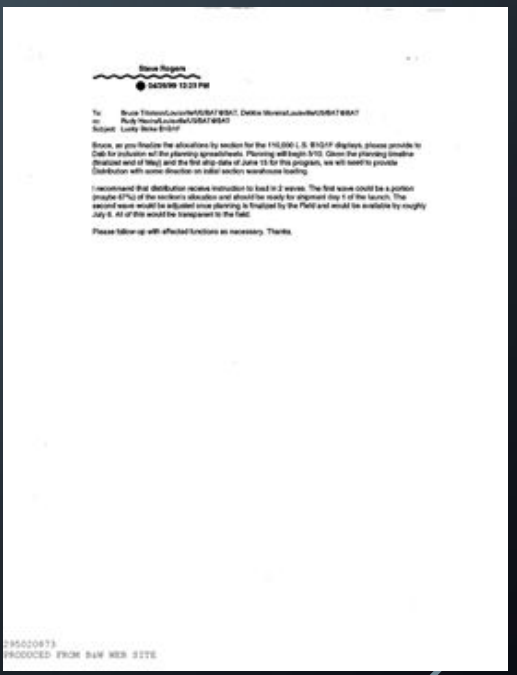
# Other Experimentation

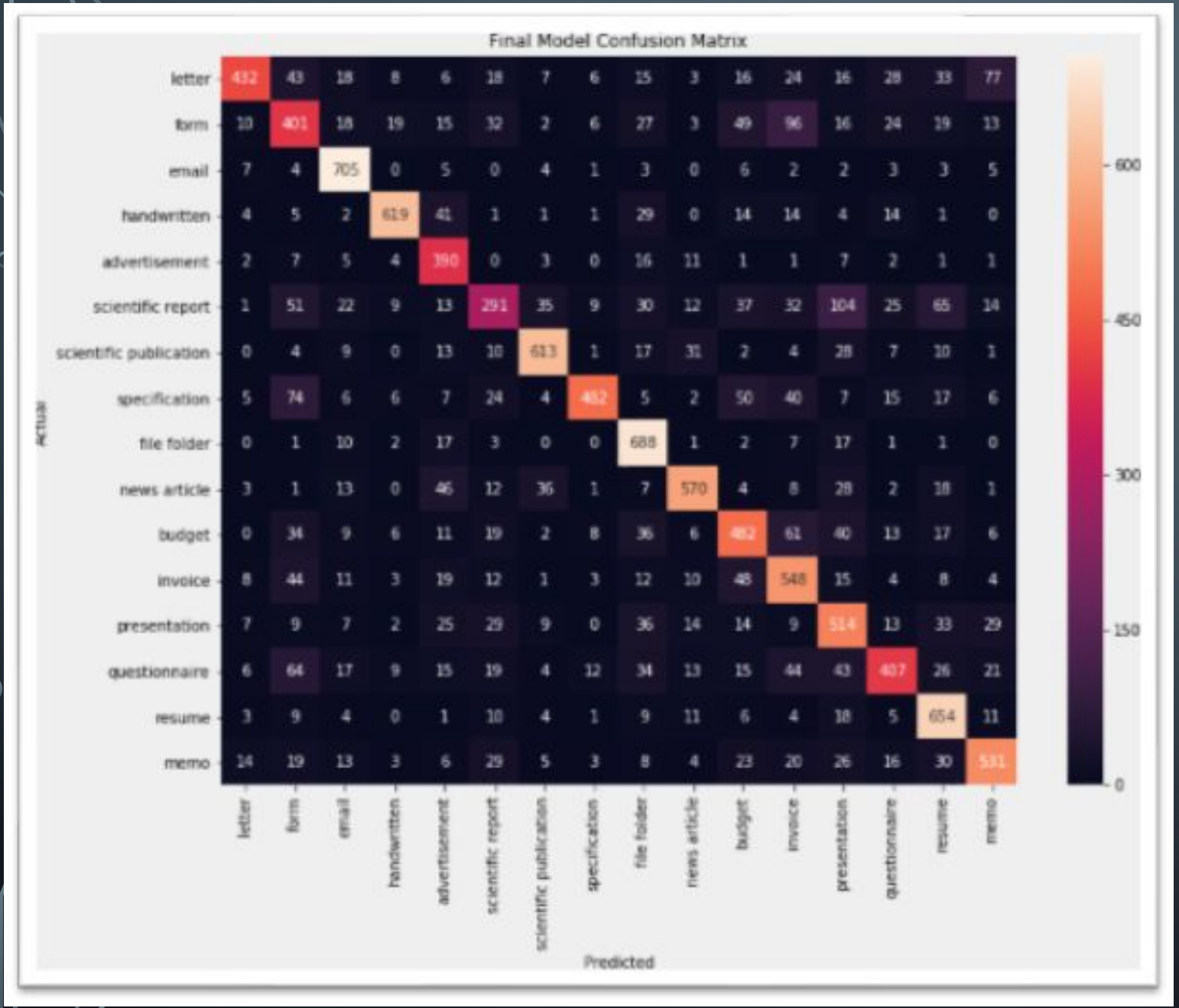
| REGION | ACCURACY | RECALL | PRECISION | F1 |
|---|---|---|---|---|
| CENTER | 0.281 | 0.288 | 0.295 | 0.268 |
| HEADER | 0.167 | 0.179 | 0.426 | 0.426 |

| POOLING TYPE | ACCURACY | RECALL | PRECISION | F1 |
|---|---|---|---|---|
| AVERAGE | 0.304 | 0.308 | 0.469 | 0.279 |
| MAX | 0.728 | 0.729 | 0.736 | 0.7234 |

Both Emails



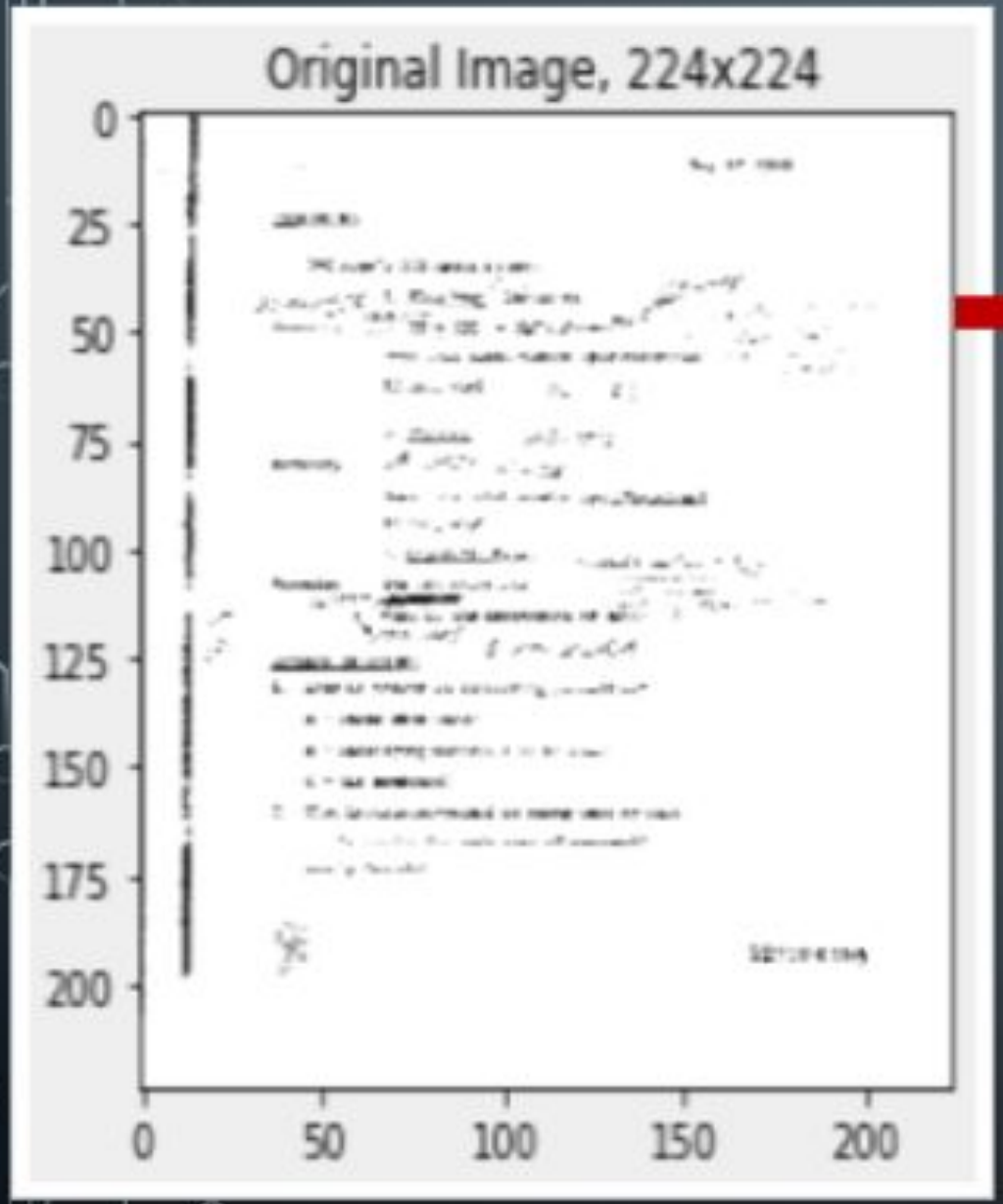


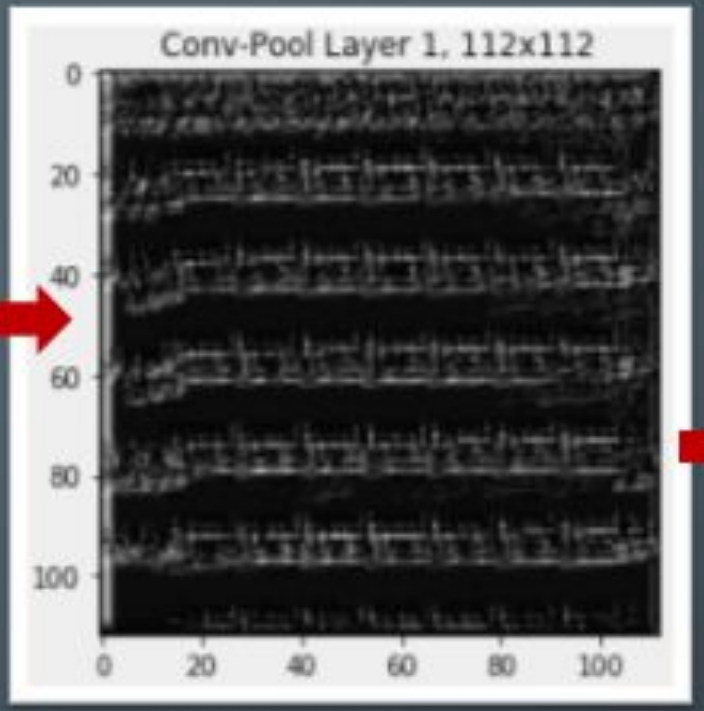


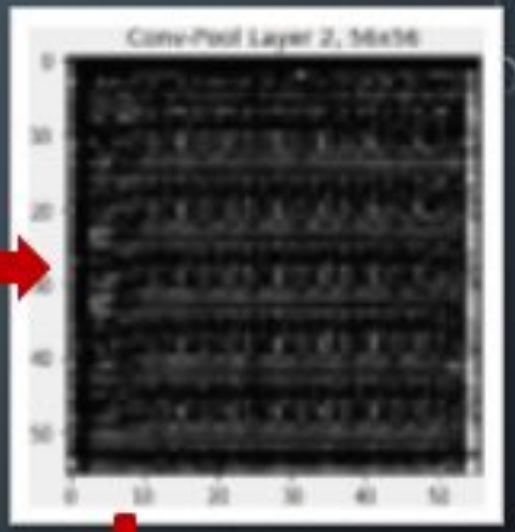

| | |
|---|---|
| **Accuracy** | 0.728 |
| **Recall** | 0.729 |
| **Precision** | 0.736 |
| **F1** | 0.7234 |

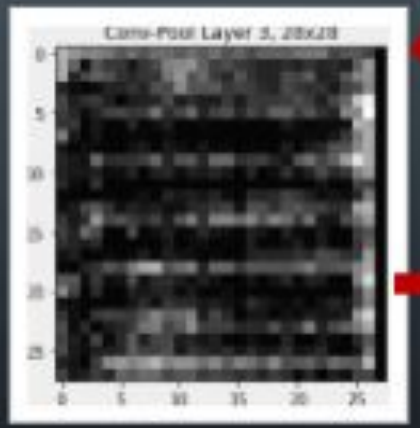
Architecture:

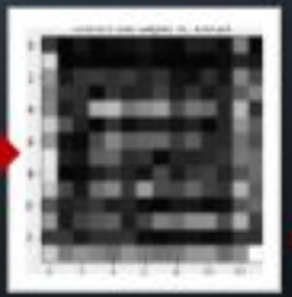
- Hist Equalization
- 224x224 Images
- 5 Conv Blocks
- Two FC w/ Dropout
- 7-5-5-3-3 Kernels
- 50 Batch Size
- 0.001 Learning Rate
- Full Image


Original Image, 224x224
0
25
50
75
100
125
150
175
200
0
50
100
150
200


Conv-Pool Layer 1, 112x112
0
20
40
60
80
100
0
20
40
60
80
100


Conv-Pool Layer 2, 56x56


Conv-Pool Layer 3, 28x28



# Keras

1. Preprocessing
2. Convolution Neural Network
3. VGG16

# Preprocessing

- convert_4darray: Function which takes image as input, rescales the image as 224*224 pixels and then converts it into a 4D array .
- convert_4darrays: Function which takes image paths as input and returns a 4D array of images using convert_4d array function.
- The images are rescaled by dividing every pixel in every image by 255.

# Convolution Neural Network

- Header: Train images: 144 per class, Test and validation images: 120 per class
- Whole: Train images: 137 per class, Test and validation images: 60 per class
- epochs: 10
- optimizer: adam
- batch_size: 32, 64
- Result: Very low accuracy

| Layer (type) | Output Shape | Param # |
|---|---|---|
| conv2d_1 (Conv2D) | (None, 224, 224, 32) | 4736 |
| max_pooling2d_1 (MaxPooling2 | (None, 74, 74, 32) | 0 |
| conv2d_2 (Conv2D) | (None, 74, 74, 32) | 25632 |
| max_pooling2d_2 (MaxPooling2 | (None, 24, 24, 32) | 0 |
| conv2d_3 (Conv2D) | (None, 24, 24, 32) | 9248 |
| max_pooling2d_3 (MaxPooling2 | (None, 8, 8, 32) | 0 |
| conv2d_4 (Conv2D) | (None, 8, 8, 32) | 9248 |
| conv2d_5 (Conv2D) | (None, 8, 8, 32) | 9248 |
| max_pooling2d_4 (MaxPooling2 | (None, 2, 2, 32) | 0 |
| flatten_1 (Flatten) | (None, 128) | 0 |
| dense_1 (Dense) | (None, 32) | 4128 |
| dense_2 (Dense) | (None, 32) | 1056 |
| dense_3 (Dense) | (None, 16) | 528 |

Total params: 63,824
Trainable params: 63,824
Non-trainable params: 0

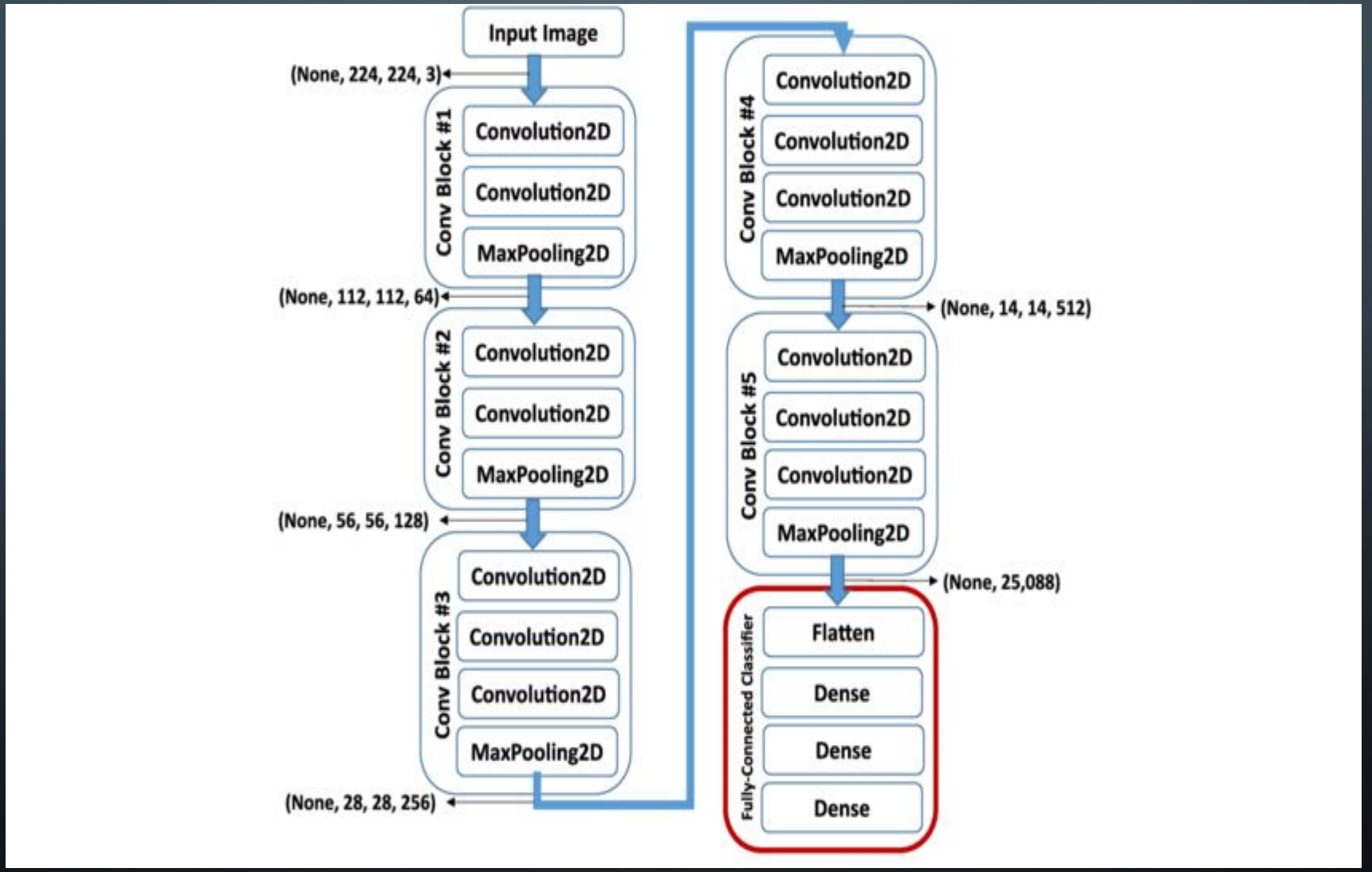
# Transfer Learning

- Process of taking a pre-trained model and “fine-tuning” the model with new dataset.
- VGG16 is used  in this dataset.

# VGG16

- Also called Oxfordnet
- Named after Visual Geometry Group of Oxford
- 16 layer deep network
- Can classify upto 1000 images
- imported from keras.applications
  - contains models with weights trained on Imagenet

# VGG16 Architecture



# VGG16(include_top = False, weights = 'imagenet', input_tensor = None, input_shape = None, pooling = None, classes= 1000)

Arguments:

1. include_top: decides whether to include 3-fully connected layers at the top of the network
2. weights:
    a. None: the weights are randomly initialized.
    b. imagenet: uses weights form pre-trained Imagenet.
3. input _tensor: optional. This is output of layers.Input()
4. input_shape: optional.

    Only specified if include_top is specified as False. Else the input_shape has to be 224*224 which channels 3.
5. pooling: optional. Used for feature extraction when include_top is specified as False.
    a. None:
    b. avg
    c. max:
6. classes: optional. Only to be specified if include_top is False, and weights is specified as 'None'.

    Specifies number of classes to classify images into.

# VGG16

- Header: Train images: 144 per class, Test and validation images: 120 per class
- Whole: Train images: 137 per class, Test and validation images: 60 per class
- optimizers used: SGD, RMSprop, Adam
  - Adam has good accuracy compared to other two optimizers
- batch_size: 32, 64
- Accuracy: 66.67
- Improved accuracy compared to previous CNN model.

# Future Directions

- Train the model in such a way that when given an input image the model should classify the image automatically.
- Read data from the document images
- Reduce class separation and combine into more homogeneous classes
- Create top-5 classification for less certain predictions

# Conclusions

- Document images more difficult to identify than natural images
- High intraclass variance makes identification even more difficult
- PyTorch produced the best results
- Transfer learning such a VGG may not be as good as custom network for non-natural imagery. Needs more research.