**Dataset**

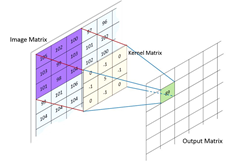
# The data will come the RVL-CDIP Dataset (Ryerson Vision Lab Complex Document Information Processing). This dataset consists of 400,000 greyscale images of 16 classes of documents. The images are presplit into 320,000 training images, 40,000 validation images, and 40,000 testing images. All image dimensions are 1000 pixels or less, though this will be scaled down for training efficiency. The data consists of 16 document classes: letter, form, email, handwritten, advertisement, scientific report, scientific publication, specification, file folder, news article, budget, invoice, presentation, questionnaire, resume, and memo.

# This dataset is sufficiently large to train a deep neural network. With 320,000 training images and 16 classes, the dataset provides roughly 20,000 images per class, far more than need to effectively train a neural network efficiently. For the sake of efficiency, random class subsets will be taken from the training, validation, and testing datasets to scale down the overall data size while maintaining class proportions.

**Input Layer**

The input layer is a three-dimensional matrix, For image analysis, the dimensions represent image height, width, and number of color channels, and the values represent pixel values within the range [0, 255]. Grayscale images have a single color channel while color images have three color channels, one each for red, blue, and green.

**Convolutional Layers**

The convolution layer consists of a set of learnable kernels (also called filters). The kernels are matrices of randomly initialized weight values in a 3-dimensional matrix (height, width, number of color channels).. In the feedforward process, the kernels slide (convolve) over each element of the input matrix and compute the sum of the element-wise multiplication between the kernel and the local image pixel values. This results in a single value that maps to the new feature map. 

Each feature map is the resulting matrix from the computed values of a single kernel convolving over the whole of the input matrix, with one feature map created for each kernel.

Aside from the number of kernels, convolutional layers have several adjustable parameters that control output size. Padding is a parameter that is used to maintain input size. The process of convolution results in a single value at the center of the kernel, which means that the edges of the input are cut off (*Figure)* and the output is *n-1* (n representing the height and width of the kernel) dimensionssmaller than the input. To prevent this, layers of zeros or other values are added around outside of the input so that the first position of the kernel is centered on the upper-left pixel value. The number of padding layers is calculated by *(n-1)/2* with n representing kernel size.

**ReLU Activation**

**Batch Normalization**

**Max Pooling**

**Flattening Layer**

**Fully Connected Layers**

**Softmax Activation**

**Loss**

**Optimization**



