## Implementation of CNN for Image Classification without Hyperparameter Tunning

| Stages of Sequential Model | Type of Layer | Number of Units | Size | Activation Function |
|---|---|---|---|---|
| First | Convolutional Layer | 32 Filters | 3X3 Kernel size | ReLU |
| Second | Max-pooling | -- | 2X2 Pooling window | None |
| Third | Convolutional Layer | 64 Filters | 3X3 Kernel size | ReLU |
| Fourth | Max-pooling | -- | 2X2 Pooling window | None |
| Fifth | Convolutional Layer | 64 Filters | 3X3 Kernel size | ReLU |
| Sixth | Flattening | -- | -- | -- |
| Seventh | Dense (Fully-Connected) Layer | 64 units | -- | ReLU |
| Eight | Dense (Fully-Connected) Layer | 10 units | -- | Softmax |
| Optimizer | Adam | Default learning rate | -- | Not Applicable |

# Implementation of CNN for Image Classification with Basic Hyperparameter Tunning

| Stages of Sequential Model | Type of Layer | Number of Units | Size | Activation Function |
|---|---|---|---|---|
| First | Convolutional Layer | 32 to 128 (Step size = 16) | 3X3 Kernel size | ReLU |
| Second | Max-pooling | -- | 2X2 Pooling window | Not Applicable |
| Third | Convolutional Layer | 32 to 128 (Step size = 16) | 3X3 Kernel size | ReLU |
| Fourth | Max-pooling | -- | 2X2 Pooling window | Not Applicable |
| Fifth | Convolutional Layer | 32 to 128 (Step size = 16) | 3X3 Kernel size | ReLU |
| Sixth | Flattening | -- | -- | Not Applicable |
| Seventh | Dense (Fully-Connected) Layer | 32 to 128 (Step size = 16) | -- | ReLU |
| Eight | Dropout Layer | Dropout Rate = 0% to 50% (Step size = 10%) | -- | Not Applicable |
| Nineth | Dense (Fully-Connected) Layer | 10 units | -- | Softmax |
| Optimizer | Adam | Learning Rate = 0.01, 0.001, 0.0001 (to select between these 3 choices) | -- | Not Applicable |

## Implementation of CNN for Image Classification with Adv. Hyperparameter Tunning

**Tune the following hyperparameters (by letting the tuner select the):**

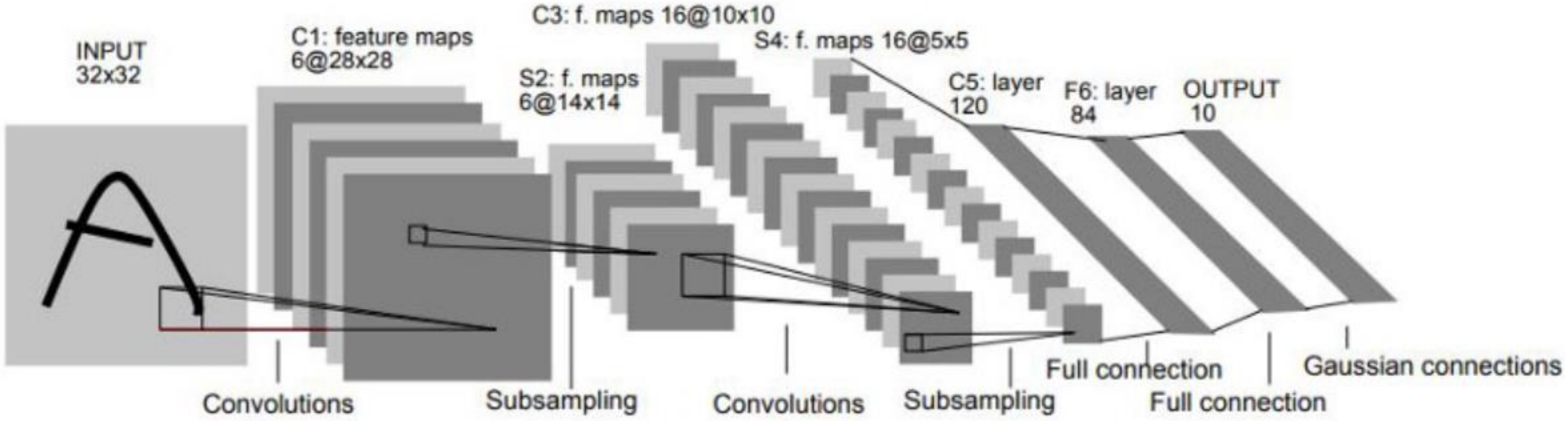
1. Number of Convolutional Layers (1,2 or 3 layers)
2. Number of filters in each Convolutional layer (32 to 128, step size of 16)

   filters: 32, 48, 64, 80, 96, 112, 128
3. Number of Dense Layers (1,2 or 3 layers)
4. Number of units in each Dense layer (32 to 128, step size of 16)

   units: 32, 48, 64, 80, 96, 112, 128
5. The last Dense layer should have output 10 units
6. The Dropout Rate (0.0 to 0.5, step size of 0.1)

   dropout rate: 0.0, 0.1, 0.2, 0.3, 0.4, 0.5
7. The Learning Rate (to choose between: 0.01, 0.001, 0.0001)

# Implementation of Image Classification using LeNet-5 CNN Architecture



## Implementation of Image Classification using LeNet-5 CNN Architecture

| Stages of the CNN Model | Type of Layer | Number of Units | Size | Activation Function |
|---|---|---|---|---|
| First | Convolutional Layer | 6 Filters | 5X5 Kernel size | tanh |
| Second | Average Pooling | -- | 2X2 Pooling window | None |
| Third | Convolutional Layer | 16 Filters | 5X5 Kernel size | tanh |
| Fourth | Average Pooling | -- | 2X2 Pooling window | None |
| Fifth | Convolutional Layer | 120 Filters | 5X5 Kernel size | tanh |
| Sixth | Flattening | -- | -- | -- |
| Seventh | Dense (Fully-Connected) Layer | 84 units | -- | tanh |
| Eight | Dense (Fully-Connected) Layer | 10 units | -- | Softmax |
| Optimizer | Adam<br>With loss function:<br>categorical_crossentropy | Default learning rate | -- | Not Applicable |