

# Laporan Project UAS Machine Learning

## Kelompok:

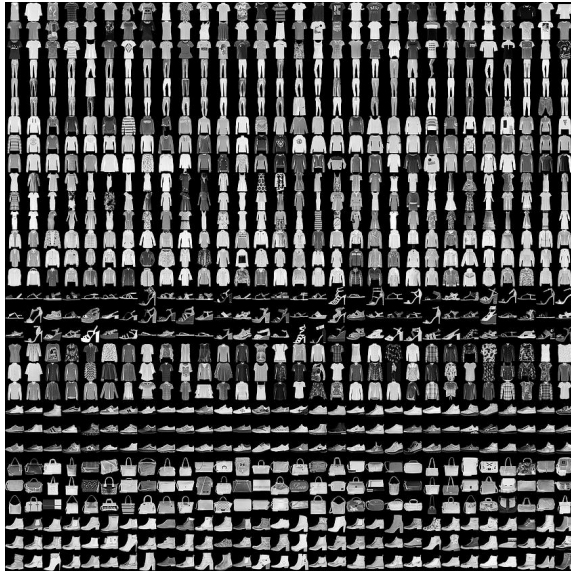
Leonardo Marcellino Gunawan - C14210030

Jason Christopher - C14210098

Michael Putra Handidjaja - C14210184

## Dataset

Fashion MNIST



[Fashion-MNIST Dataset | Papers With Code](#)

Fashion-MNIST is a dataset comprising of  $28 \times 28$  grayscale images of 70,000 fashion products from 10 categories, with 7,000 images per category. The training set has 60,000 images and the test set has 10,000 images. Fashion-MNIST shares the same image size, data format and the structure of training and testing splits with the original MNIST.

# Data Preprocessing

## Check Null

```
# Check for null values in X_train
if np.any(np.isnan(X_train)):
    print("Null values found in X_train.")
else:
    print("No null values in X_train.")

# Check for null values in y_train
if np.any(np.isnan(y_train)):
    print("Null values found in y_train.")
else:
    print("No null values in y_train.")

# Check for null values in X_test
if np.any(np.isnan(X_test)):
    print("Null values found in X_test.")
else:
    print("No null values in X_test.")

# Check for null values in y_test
if np.any(np.isnan(y_test)):
    print("Null values found in y_test.")
else:
    print("No null values in y_test.")
```

```
No null values in X_train.
```

```
No null values in y_train.
```

```
No null values in X_test.
```

```
No null values in y_test.
```

## Check the data distribution

```
classes = ['T-shirt/top', 'Trouser', 'Pullover', 'Dress', 'Coat',
'Sandal', 'Shirt', 'Sneaker', 'Bag', 'Ankle boot']

# Count the occurrences of each class in y_train
train_class_counts = np.bincount(y_train)

# Count the occurrences of each class in y_test
```

```

test_class_counts = np.bincount(y_test)

# Print the data distribution for y_train with labels
print("Data Distribution for y_train:")
for i in range(len(classes)):
    print(f'{classes[i]}: {train_class_counts[i]}')

# Print the data distribution for y_test with labels
print("\nData Distribution for y_test:")
for i in range(len(classes)):
    print(f'{classes[i]}: {test_class_counts[i]}')

```

```

Data Distribution for y_train:
T-shirt/top: 6000
Trouser: 6000
Pullover: 6000
Dress: 6000
Coat: 6000
Sandal: 6000
Shirt: 6000
Sneaker: 6000
Bag: 6000
Ankle boot: 6000

```

```

Data Distribution for y_test:
T-shirt/top: 1000
Trouser: 1000
Pullover: 1000
Dress: 1000
Coat: 1000
Sandal: 1000
Shirt: 1000
Sneaker: 1000
Bag: 1000
Ankle boot: 1000

```

Shuffle dan Normalize data supaya range nya menjadi 0-1

```

from sklearn.utils import shuffle

# Shuffle the training data
X_train, y_train = shuffle(X_train, y_train, random_state=42)
X_test, y_test = shuffle(X_test, y_test, random_state=42)
X_train = X_train / 255
X_test = X_test / 255

```

# Metode CNN

Metode CNN menggunakan AlexNet architecture dengan adaptive learning rate.

## Parameter Configuration

Configuration ini dijalankan dengan 15 epochs semua dan menggunakan loss `sparse_categorical_crossentropy` karena output datanya berupa angka 0-9 (yang merupakan indikasi dari class-classnya)

Configurati on	Layers	Regularization	Optimizer	Train Error	Test Error
Conf1	<div>1. Convolutional Layer 1:<ul style="list-style-type: none"><li>- Filters: 32</li><li>- Kernel Size: (11, 11)</li><li>- Activation: ReLU</li><li>- Input Shape: (28, 28, 1)</li></ul></div> <div>2. Batch Normalization Layer 1:<ul style="list-style-type: none"><li>- Applies batch normalization after Convolutional Layer 1.</li></ul></div> <div>3. Max Pooling Layer 1:<ul style="list-style-type: none"><li>- Pool Size: (3, 3)</li><li>- Strides: 1</li></ul></div> <div>4. Convolutional Layer 2:<ul style="list-style-type: none"><li>- Filters: 64</li><li>- Kernel Size: (5, 5)</li><li>- Activation: ReLU</li><li>- Padding: 'same'</li></ul></div> <div>5. Batch Normalization Layer 2:<ul style="list-style-type: none"><li>- Applies batch normalization after Convolutional Layer 2.</li></ul></div> <div>6. Max Pooling Layer 2:<ul style="list-style-type: none"><li>- Pool Size: (3, 3)</li><li>- Strides: 1</li></ul></div>	<div>1. Dropout Layer 1:<ul style="list-style-type: none"><li>- Rate: 0.25</li><li>- Seed: 42</li></ul></div> <div>2. Dropout Layer 2:<ul style="list-style-type: none"><li>- Rate: 0.25</li><li>- Seed: 42</li></ul></div>	SGD momentum 0.9	0.2498	0.2757

	<p>7. Convolutional Layers 3, 4, 5:</p> <ul style="list-style-type: none"> <li>- Convolutional Layer 3: <ul style="list-style-type: none"> <li>- Filters: 64</li> <li>- Kernel Size: (5, 5)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> </li> <li>- Convolutional Layer 4: <ul style="list-style-type: none"> <li>- Filters: 64</li> <li>- Kernel Size: (5, 5)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> </li> <li>- Convolutional Layer 5: <ul style="list-style-type: none"> <li>- Filters: 32</li> <li>- Kernel Size: (5, 5)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> </li> </ul> <p>8. Max Pooling Layer 3:</p> <ul style="list-style-type: none"> <li>- Pool Size: (3, 3)</li> <li>- Strides: 1</li> </ul> <p>9. Flatten Layer:</p> <ul style="list-style-type: none"> <li>- Flattens the input to prepare for the fully connected layers.</li> </ul> <p>10. Dense Layer 1:</p> <ul style="list-style-type: none"> <li>- Units: 64</li> <li>- Activation: ReLU</li> <li>- Weight Initialization: GlorotUniform</li> <li>- Batch Normalization</li> </ul> <p>11. Dense Layer 2:</p> <ul style="list-style-type: none"> <li>- Units: 64</li> <li>- Activation: ReLU</li> <li>- Weight Initialization: GlorotUniform</li> <li>- Batch Normalization</li> </ul> <p>12. Output Dense Layer:</p> <ul style="list-style-type: none"> <li>- Units: 10 (for 10 classes in classification)</li> <li>- Activation: Softmax</li> <li>- Weight Initialization: GlorotUniform</li> </ul>				
--	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--	--	--	--

Conf2	<p>1. Convolutional Layer 1:</p> <ul style="list-style-type: none"> <li>- Filters: 32</li> <li>- Kernel Size: (11, 11)</li> <li>- Activation: ReLU</li> <li>- Input Shape: (28, 28, 1)</li> </ul> <p>2. Batch Normalization Layer 1:</p> <ul style="list-style-type: none"> <li>- Applies batch normalization after Convolutional Layer 1.</li> </ul> <p>3. Max Pooling Layer 1:</p> <ul style="list-style-type: none"> <li>- Pool Size: (3, 3)</li> <li>- Strides: 1</li> </ul> <p>4. Convolutional Layer 2:</p> <ul style="list-style-type: none"> <li>- Filters: 64</li> <li>- Kernel Size: (5, 5)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> <p>5. Batch Normalization Layer 2:</p> <ul style="list-style-type: none"> <li>- Applies batch normalization after Convolutional Layer 2.</li> </ul> <p>6. Max Pooling Layer 2:</p> <ul style="list-style-type: none"> <li>- Pool Size: (3, 3)</li> <li>- Strides: 1</li> </ul> <p>7. Convolutional Layers 3, 4, 5:</p> <ul style="list-style-type: none"> <li>- Convolutional Layer 3: <ul style="list-style-type: none"> <li>- Filters: 64</li> <li>- Kernel Size: (5, 5)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> </li> <li>- Convolutional Layer 4: <ul style="list-style-type: none"> <li>- Filters: 64</li> <li>- Kernel Size: (5, 5)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> </li> <li>- Convolutional Layer 5: <ul style="list-style-type: none"> <li>- Filters: 32</li> <li>- Kernel Size: (5, 5)</li> <li>- Activation: ReLU</li> </ul> </li> </ul>	<p>1. Dropout Layer 1:</p> <ul style="list-style-type: none"> <li>- Rate: 0.25</li> <li>- Seed: 42</li> </ul> <p>2. Dropout Layer 2:</p> <ul style="list-style-type: none"> <li>- Rate: 0.25</li> <li>- Seed: 42</li> </ul>	SGD momentum 0.9		
-------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	------------------	--	--

	<ul style="list-style-type: none"> <li>- Padding: 'same'</li> </ul> <p>8. Max Pooling Layer 3:</p> <ul style="list-style-type: none"> <li>- Pool Size: (3, 3)</li> <li>- Strides: 1</li> </ul> <p>9. Flatten Layer:</p> <ul style="list-style-type: none"> <li>- Flattens the input to prepare for the fully connected layers.</li> </ul> <p>10. Dense Layer 1:</p> <ul style="list-style-type: none"> <li>- Units: 64</li> <li>- Activation: ReLU</li> <li>- Weight Initialization: GlorotUniform</li> <li>- Batch Normalization</li> </ul> <p>11. Dense Layer 2:</p> <ul style="list-style-type: none"> <li>- Units: 64</li> <li>- Activation: ReLU</li> <li>- Weight Initialization: GlorotUniform</li> <li>- Batch Normalization</li> </ul> <p>12. Output Dense Layer:</p> <ul style="list-style-type: none"> <li>- Units: 10 (for 10 classes in classification)</li> <li>- Activation: Softmax</li> <li>- Weight Initialization: GlorotUniform</li> </ul>				
Conf3	<p>1. Convolutional Layer 1:</p> <ul style="list-style-type: none"> <li>- Filters: 32</li> <li>- Kernel Size: (11, 11)</li> <li>- Activation: ReLU</li> <li>- Input Shape: (28, 28, 1)</li> </ul> <p>2. Batch Normalization Layer 1:</p> <ul style="list-style-type: none"> <li>- Applies batch normalization after Convolutional Layer 1.</li> </ul> <p>3. Max Pooling Layer 1:</p> <ul style="list-style-type: none"> <li>- Pool Size: (3, 3)</li> <li>- Strides: 1</li> </ul> <p>4. Convolutional Layer 2:</p>	<p>1. Dropout Layer 1:</p> <ul style="list-style-type: none"> <li>- Rate: 0.25</li> <li>- Seed: 42</li> </ul> <p>2. Dropout Layer 2:</p> <ul style="list-style-type: none"> <li>- Rate: 0.25</li> <li>- Seed: 42</li> </ul>	SGD momentum 0.9	0.2352	0.2714

	<ul style="list-style-type: none"> <li>- Filters: 64</li> <li>- Kernel Size: (5, 5)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> <p>5. Batch Normalization Layer 2:</p> <ul style="list-style-type: none"> <li>- Applies batch normalization after Convolutional Layer 2.</li> </ul> <p>6. Max Pooling Layer 2:</p> <ul style="list-style-type: none"> <li>- Pool Size: (3, 3)</li> <li>- Strides: 1</li> </ul> <p>7. Convolutional Layers 3, 4, 5:</p> <ul style="list-style-type: none"> <li>- Convolutional Layer 3: <ul style="list-style-type: none"> <li>- Filters: 64</li> <li>- Kernel Size: (5, 5)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> </li> <li>- Convolutional Layer 4: <ul style="list-style-type: none"> <li>- Filters: 64</li> <li>- Kernel Size: (5, 5)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> </li> <li>- Convolutional Layer 5: <ul style="list-style-type: none"> <li>- Filters: 32</li> <li>- Kernel Size: (5, 5)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> </li> </ul> <p>8. Max Pooling Layer 3:</p> <ul style="list-style-type: none"> <li>- Pool Size: (3, 3)</li> <li>- Strides: 1</li> </ul> <p>9. Flatten Layer:</p> <ul style="list-style-type: none"> <li>- Flattens the input to prepare for the fully connected layers.</li> </ul> <p>10. Dense Layer 1:</p> <ul style="list-style-type: none"> <li>- Units: 64</li> <li>- Activation: ReLU</li> <li>- Weight Initialization: GlorotUniform</li> <li>- Batch Normalization</li> </ul> <p>11. Dense Layer 2:</p>				
--	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--	--	--	--



	<ul style="list-style-type: none"> <li>- Units: 32</li> <li>- Activation: ReLU</li> <li>- Weight Initialization: GlorotUniform</li> <li>- Batch Normalization</li> </ul> <p>12. Output Dense Layer:</p> <ul style="list-style-type: none"> <li>- Units: 10 (for 10 classes in classification)</li> <li>- Activation: Softmax</li> <li>- Weight Initialization: GlorotUniform</li> </ul>				
Conf4	<p>1. Convolutional Layer 1:</p> <ul style="list-style-type: none"> <li>- Filters: 32</li> <li>- Kernel Size: (11, 11)</li> <li>- Activation: ReLU</li> <li>- Input Shape: (28, 28, 1)</li> </ul> <p>2. Batch Normalization Layer 1:</p> <ul style="list-style-type: none"> <li>- Applies batch normalization after Convolutional Layer 1.</li> </ul> <p>3. Max Pooling Layer 1:</p> <ul style="list-style-type: none"> <li>- Pool Size: (3, 3)</li> <li>- Strides: 1</li> </ul> <p>4. Convolutional Layer 2:</p> <ul style="list-style-type: none"> <li>- Filters: 64</li> <li>- Kernel Size: (7, 7)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> <p>5. Batch Normalization Layer 2:</p> <ul style="list-style-type: none"> <li>- Applies batch normalization after Convolutional Layer 2.</li> </ul> <p>6. Max Pooling Layer 2:</p> <ul style="list-style-type: none"> <li>- Pool Size: (5, 5)</li> <li>- Strides: 2</li> </ul> <p>7. Convolutional Layers 3, 4, 5:</p> <ul style="list-style-type: none"> <li>- Convolutional Layer 3: <ul style="list-style-type: none"> <li>- Filters: 64</li> <li>- Kernel Size: (3, 3)</li> </ul> </li> </ul>	<p>1. Dropout Layer 1:</p> <ul style="list-style-type: none"> <li>- Rate: 0.25</li> <li>- Seed: 42</li> </ul> <p>2. Dropout Layer 2:</p> <ul style="list-style-type: none"> <li>- Rate: 0.25</li> <li>- Seed: 42</li> </ul>	SGD momentum 0.9	0.3298	0.3129

	<ul style="list-style-type: none"> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> <ul style="list-style-type: none"> <li>- Convolutional Layer 4: <ul style="list-style-type: none"> <li>- Filters: 64</li> <li>- Kernel Size: (3, 3)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> </li> <li>- Convolutional Layer 5: <ul style="list-style-type: none"> <li>- Filters: 32</li> <li>- Kernel Size: (3, 3)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> </li> </ul> <p>8. Max Pooling Layer 3:</p> <ul style="list-style-type: none"> <li>- Pool Size: (3, 3)</li> <li>- Strides: 1</li> </ul> <p>9. Flatten Layer:</p> <ul style="list-style-type: none"> <li>- Flattens the input to prepare for the fully connected layers.</li> </ul> <p>10. Dense Layer 1:</p> <ul style="list-style-type: none"> <li>- Units: 64</li> <li>- Activation: ReLU</li> <li>- Weight Initialization: GlorotUniform</li> <li>- Batch Normalization</li> </ul> <p>11. Dense Layer 2:</p> <ul style="list-style-type: none"> <li>- Units: 32</li> <li>- Activation: ReLU</li> <li>- Weight Initialization: GlorotUniform</li> <li>- Batch Normalization</li> </ul> <p>12. Output Dense Layer:</p> <ul style="list-style-type: none"> <li>- Units: 10 (for 10 classes in classification)</li> <li>- Activation: Softmax</li> <li>- Weight Initialization: GlorotUniform</li> </ul>				
Conf5	<p>1. Convolutional Layer 1:</p> <ul style="list-style-type: none"> <li>- Filters: 32</li> <li>- Kernel Size: (11, 11)</li> <li>- Activation: ReLU</li> <li>- Input Shape: (28, 28, 1)</li> </ul> <p>2. Batch Normalization</p>	<p>1. Dropout Layer 1:</p> <ul style="list-style-type: none"> <li>- Rate: 0.25</li> <li>- Seed: 42</li> </ul> <p>2. Dropout Layer 2:</p>	SGD momentum 0.9	0.2093	0.2816

	<p>Layer 1:</p> <ul style="list-style-type: none"> <li>- Applies batch normalization after Convolutional Layer 1.</li> </ul> <p>3. Max Pooling Layer 1:</p> <ul style="list-style-type: none"> <li>- Pool Size: (3, 3)</li> <li>- Strides: 2</li> </ul> <p>4. Convolutional Layer 2:</p> <ul style="list-style-type: none"> <li>- Filters: 64</li> <li>- Kernel Size: (5, 5)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> <p>5. Batch Normalization Layer 2:</p> <ul style="list-style-type: none"> <li>- Applies batch normalization after Convolutional Layer 2.</li> </ul> <p>6. Max Pooling Layer 2:</p> <ul style="list-style-type: none"> <li>- Pool Size: (3, 3)</li> <li>- Strides: 2</li> </ul> <p>7. Convolutional Layers 3, 4, 5:</p> <ul style="list-style-type: none"> <li>- Convolutional Layer 3: <ul style="list-style-type: none"> <li>- Filters: 64</li> <li>- Kernel Size: (3, 3)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> </li> <li>- Convolutional Layer 4: <ul style="list-style-type: none"> <li>- Filters: 64</li> <li>- Kernel Size: (3, 3)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> </li> <li>- Convolutional Layer 5: <ul style="list-style-type: none"> <li>- Filters: 32</li> <li>- Kernel Size: (3, 3)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> </li> </ul> <p>8. Max Pooling Layer 3:</p> <ul style="list-style-type: none"> <li>- Pool Size: (3, 3)</li> <li>- Strides: 2</li> </ul> <p>9. Flatten Layer:</p> <ul style="list-style-type: none"> <li>- Flattens the input to prepare for the fully</li> </ul>	<ul style="list-style-type: none"> <li>- Rate: 0.25</li> <li>- Seed: 42</li> </ul>			
--	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------	--	--	--

	<p>connected layers.</p> <p>10. Dense Layer 1:</p> <ul style="list-style-type: none"> <li>- Units: 64</li> <li>- Activation: ReLU</li> <li>- Weight Initialization: GlorotUniform</li> <li>- Batch Normalization</li> </ul> <p>11. Dense Layer 2:</p> <ul style="list-style-type: none"> <li>- Units: 32</li> <li>- Activation: ReLU</li> <li>- Weight Initialization: GlorotUniform</li> <li>- Batch Normalization</li> </ul> <p>12. Output Dense Layer:</p> <ul style="list-style-type: none"> <li>- Units: 10 (for 10 classes in classification)</li> <li>- Activation: Softmax</li> <li>- Weight Initialization: GlorotUniform</li> </ul>				
Conf6	<p>1. Convolutional Layer 1:</p> <ul style="list-style-type: none"> <li>- Filters: 32</li> <li>- Kernel Size: (4, 4)</li> <li>- Activation: ReLU</li> <li>- Input Shape: (28, 28, 1)</li> </ul> <p>2. Batch Normalization Layer 1:</p> <ul style="list-style-type: none"> <li>- Applies batch normalization after Convolutional Layer 1.</li> </ul> <p>3. Max Pooling Layer 1:</p> <ul style="list-style-type: none"> <li>- Pool Size: (2, 2)</li> <li>- Strides: 1</li> </ul> <p>4. Convolutional Layer 2:</p> <ul style="list-style-type: none"> <li>- Filters: 64</li> <li>- Kernel Size: (5, 5)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> <p>5. Batch Normalization Layer 2:</p> <ul style="list-style-type: none"> <li>- Applies batch normalization after Convolutional Layer 2.</li> </ul>	<p>1. Dropout Layer 1:</p> <ul style="list-style-type: none"> <li>- Rate: 0.25</li> <li>- Seed: 42</li> </ul> <p>2. Dropout Layer 2:</p> <ul style="list-style-type: none"> <li>- Rate: 0.25</li> <li>- Seed: 42</li> </ul>	SGD momentum 0.9	0.1549	0.2672

	<p>6. Max Pooling Layer 2:</p> <ul style="list-style-type: none"> <li>- Pool Size: (2,2)</li> <li>- Strides: 2</li> </ul> <p>7. Convolutional Layers 3, 4, 5:</p> <ul style="list-style-type: none"> <li>- Convolutional Layer 3: <ul style="list-style-type: none"> <li>- Filters: 64</li> <li>- Kernel Size: (3, 3)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> </li> <li>- Convolutional Layer 4: <ul style="list-style-type: none"> <li>- Filters: 64</li> <li>- Kernel Size: (3, 3)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> </li> <li>- Convolutional Layer 5: <ul style="list-style-type: none"> <li>- Filters: 32</li> <li>- Kernel Size: (3, 3)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> </li> </ul> <p>8. Max Pooling Layer 3:</p> <ul style="list-style-type: none"> <li>- Pool Size: (2, 2)</li> <li>- Strides: 1</li> </ul> <p>9. Flatten Layer:</p> <ul style="list-style-type: none"> <li>- Flattens the input to prepare for the fully connected layers.</li> </ul> <p>10. Dense Layer 1:</p> <ul style="list-style-type: none"> <li>- Units: 128</li> <li>- Activation: ReLU</li> <li>- Weight Initialization: GlorotUniform</li> <li>- Batch Normalization</li> </ul> <p>11. Dense Layer 2:</p> <ul style="list-style-type: none"> <li>- Units: 64</li> <li>- Activation: ReLU</li> <li>- Weight Initialization: GlorotUniform</li> <li>- Batch Normalization</li> </ul> <p>12. Output Dense Layer:</p> <ul style="list-style-type: none"> <li>- Units: 10 (for 10 classes in classification)</li> <li>- Activation: Softmax</li> </ul>				
--	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--	--	--	--

	- Weight Initialization: GlorotUniform				
Conf7	<p>1. Convolutional Layer 1:</p> <ul style="list-style-type: none"> <li>- Filters: 32</li> <li>- Kernel Size: (5, 5)</li> <li>- Activation: ReLU</li> <li>- Input Shape: (28, 28, 1)</li> </ul> <p>2. Batch Normalization Layer 1:</p> <ul style="list-style-type: none"> <li>- Applies batch normalization after Convolutional Layer 1.</li> </ul> <p>3. Max Pooling Layer 1:</p> <ul style="list-style-type: none"> <li>- Pool Size: (5, 5)</li> <li>- Strides: 2</li> </ul> <p>4. Convolutional Layer 2:</p> <ul style="list-style-type: none"> <li>- Filters: 64</li> <li>- Kernel Size: (5, 5)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> <p>5. Batch Normalization Layer 2:</p> <ul style="list-style-type: none"> <li>- Applies batch normalization after Convolutional Layer 2.</li> </ul> <p>6. Max Pooling Layer 2:</p> <ul style="list-style-type: none"> <li>- Pool Size: (5,5)</li> <li>- Strides: 2</li> </ul> <p>7. Convolutional Layers 3, 4, 5:</p> <ul style="list-style-type: none"> <li>- Convolutional Layer 3: <ul style="list-style-type: none"> <li>- Filters: 64</li> <li>- Kernel Size: (5, 5)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> </li> <li>- Convolutional Layer 4: <ul style="list-style-type: none"> <li>- Filters: 64</li> <li>- Kernel Size: (5, 5)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> </li> <li>- Convolutional Layer 5: <ul style="list-style-type: none"> <li>- Filters: 32</li> <li>- Kernel Size: (5, 5)</li> </ul> </li> </ul>	<p>1. Dropout Layer 1:</p> <ul style="list-style-type: none"> <li>- Rate: 0.25</li> <li>- Seed: 42</li> </ul> <p>2. Dropout Layer 2:</p> <ul style="list-style-type: none"> <li>- Rate: 0.25</li> <li>- Seed: 42</li> </ul>	SGD momentum 0.9	0.1057	0.2574

	<ul style="list-style-type: none"> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> <p>8. Max Pooling Layer 3:</p> <ul style="list-style-type: none"> <li>- Pool Size: (2, 2)</li> <li>- Strides: 2</li> </ul> <p>9. Flatten Layer:</p> <ul style="list-style-type: none"> <li>- Flattens the input to prepare for the fully connected layers.</li> </ul> <p>10. Dense Layer 1:</p> <ul style="list-style-type: none"> <li>- Units: 128</li> <li>- Activation: ReLU</li> <li>- Weight Initialization: GlorotUniform</li> <li>- Batch Normalization</li> </ul> <p>11. Dense Layer 2:</p> <ul style="list-style-type: none"> <li>- Units: 64</li> <li>- Activation: ReLU</li> <li>- Weight Initialization: GlorotUniform</li> <li>- Batch Normalization</li> </ul> <p>12. Output Dense Layer:</p> <ul style="list-style-type: none"> <li>- Units: 10 (for 10 classes in classification)</li> <li>- Activation: Softmax</li> <li>- Weight Initialization: GlorotUniform</li> </ul>				
Conf8	<p>1. Convolutional Layer 1:</p> <ul style="list-style-type: none"> <li>- Filters: 64</li> <li>- Kernel Size: (5, 5)</li> <li>- Activation: ReLU</li> <li>- Input Shape: (28, 28, 1)</li> </ul> <p>2. Batch Normalization Layer 1:</p> <ul style="list-style-type: none"> <li>- Applies batch normalization after Convolutional Layer 1.</li> </ul> <p>3. Max Pooling Layer 1:</p> <ul style="list-style-type: none"> <li>- Pool Size: (3, 3)</li> </ul>	<p>1. Dropout Layer 1:</p> <ul style="list-style-type: none"> <li>- Rate: 0.25</li> <li>- Seed: 42</li> </ul>	SGD momentum 0.9	0.194	0.2481

	<ul style="list-style-type: none"> <li>- Strides: 1</li> </ul> <p>4. Convolutional Layer 2:</p> <ul style="list-style-type: none"> <li>- Filters: 32</li> <li>- Kernel Size: (5, 5)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> <p>5. Batch Normalization Layer 2:</p> <ul style="list-style-type: none"> <li>- Applies batch normalization after Convolutional Layer 2.</li> </ul> <p>6. Max Pooling Layer 2:</p> <ul style="list-style-type: none"> <li>- Pool Size: (3, 3)</li> <li>- Strides: 2</li> </ul> <p>7. Convolutional Layers 3, 4, 5:</p> <ul style="list-style-type: none"> <li>- Convolutional Layer 3:</li> <li>- Filters: 64</li> <li>- Kernel Size: (5, 5)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> <li>- Convolutional Layer 4:</li> <li>- Filters: 64</li> <li>- Kernel Size: (5, 5)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> <li>- Convolutional Layer 5:</li> <li>- Filters: 32</li> <li>- Kernel Size: (5, 5)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> <p>8. Max Pooling Layer 3:</p> <ul style="list-style-type: none"> <li>- Pool Size: (3, 3)</li> <li>- Strides: 1</li> </ul> <p>9. Flatten Layer:</p> <ul style="list-style-type: none"> <li>- Flattens the input to</li> </ul>				
--	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--	--	--	--



	<p>prepare for the fully connected layers.</p> <p>10. Dense Layer 1:</p> <ul style="list-style-type: none"> <li>- Units: 128</li> <li>- Activation: ReLU</li> <li>- Weight Initialization: GlorotUniform</li> <li>- Batch Normalization</li> </ul> <p>11. Dense Layer 2:</p> <ul style="list-style-type: none"> <li>- Units: 64</li> <li>- Activation: ReLU</li> <li>- Weight Initialization: GlorotUniform</li> <li>- Batch Normalization</li> </ul> <p>12. Output Dense Layer:</p> <ul style="list-style-type: none"> <li>- Units: 10 (for 10 classes in classification)</li> <li>- Activation: Softmax</li> <li>- Weight Initialization: GlorotUniform</li> </ul>				
Conf9	<p>1. Convolutional Layer 1:</p> <ul style="list-style-type: none"> <li>- Filters: 32</li> <li>- Kernel Size: (5, 5)</li> <li>- Activation: ReLU</li> <li>- Input Shape: (28, 28, 1)</li> </ul> <p>2. Batch Normalization Layer 1:</p> <ul style="list-style-type: none"> <li>- Applies batch normalization after Convolutional Layer 1.</li> </ul> <p>3. Max Pooling Layer 1:</p> <ul style="list-style-type: none"> <li>- Pool Size: (3, 3)</li> <li>- Strides: 1</li> </ul> <p>4. Convolutional Layer 2:</p>		SGD momentum 0.9	0.2064	0.2511

	<ul style="list-style-type: none"> <li>- Filters: 32</li> <li>- Kernel Size: (5, 5)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> <p>6. Max Pooling Layer 2:</p> <ul style="list-style-type: none"> <li>- Pool Size: (3, 3)</li> <li>- Strides: 2</li> </ul> <p>7. Convolutional Layers 3, 4, 5:</p> <ul style="list-style-type: none"> <li>- Convolutional Layer 3:</li> <li>- Filters: 32</li> <li>- Kernel Size: (5, 5)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> <li>- Convolutional Layer 4:</li> <li>- Filters: 32</li> <li>- Kernel Size: (5, 5)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> <li>- Convolutional Layer 5:</li> <li>- Filters: 32</li> <li>- Kernel Size: (5, 5)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> <p>8. Max Pooling Layer 3:</p> <ul style="list-style-type: none"> <li>- Pool Size: (3, 3)</li> <li>- Strides: 1</li> </ul> <p>9. Flatten Layer:</p> <ul style="list-style-type: none"> <li>- Flattens the input to prepare for the fully connected layers.</li> </ul> <p>10. Dense Layer 1:</p> <ul style="list-style-type: none"> <li>- Units: 64</li> <li>- Activation: ReLU</li> <li>- Weight Initialization: GlorotUniform</li> <li>- Batch Normalization</li> </ul>				
--	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--	--	--	--

	<p>11. Dense Layer 2:</p> <ul style="list-style-type: none"> <li>- Units: 64</li> <li>- Activation: ReLU</li> <li>- Weight Initialization: GlorotUniform</li> <li>- Batch Normalization</li> </ul> <p>12. Output Dense Layer:</p> <ul style="list-style-type: none"> <li>- Units: 10 (for 10 classes in classification)</li> <li>- Activation: Softmax</li> <li>- Weight Initialization: GlorotUniform</li> </ul>				
Conf10	<p>1. Convolutional Layer 1:</p> <ul style="list-style-type: none"> <li>- Filters: 32</li> <li>- Kernel Size: (7, 7)</li> <li>- Activation: ReLU</li> <li>- Input Shape: (28, 28, 1)</li> </ul> <p>2. Batch Normalization Layer 1:</p> <ul style="list-style-type: none"> <li>- Applies batch normalization after Convolutional Layer 1.</li> </ul> <p>3. Max Pooling Layer 1:</p> <ul style="list-style-type: none"> <li>- Pool Size: (3, 3)</li> <li>- Strides: 2</li> </ul> <p>4. Convolutional Layer 2:</p> <ul style="list-style-type: none"> <li>- Filters: 64</li> <li>- Kernel Size: (7, 7)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> <p>5. Batch Normalization Layer 2:</p> <ul style="list-style-type: none"> <li>- Applies batch normalization after Convolutional Layer 2.</li> </ul> <p>6. Max Pooling Layer 2:</p> <ul style="list-style-type: none"> <li>- Pool Size: (5, 5)</li> <li>- Strides: 2</li> </ul> <p>7. Convolutional Layers 3, 4, 5:</p> <ul style="list-style-type: none"> <li>- Convolutional Layer 3:</li> <ul style="list-style-type: none"> <li>- Filters: 64</li> <li>- Kernel Size: (4, 4)</li> </ul> </ul>	<p>1. Dropout Layer 1:</p> <ul style="list-style-type: none"> <li>- Rate: 0.25</li> <li>- Seed: 42</li> </ul> <p>2. Dropout Layer 2:</p> <ul style="list-style-type: none"> <li>- Rate: 0.25</li> <li>- Seed: 42</li> </ul>	SGD momentum 0.9	0.1998	0.2776

	<ul style="list-style-type: none"> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> <ul style="list-style-type: none"> <li>- Convolutional Layer 4: <ul style="list-style-type: none"> <li>- Filters: 64</li> <li>- Kernel Size: (4, 4)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> </li> <li>- Convolutional Layer 5: <ul style="list-style-type: none"> <li>- Filters: 32</li> <li>- Kernel Size: (4, 4)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> </li> </ul> <p>8. Max Pooling Layer 3:</p> <ul style="list-style-type: none"> <li>- Pool Size: (3, 3)</li> <li>- Strides: 2</li> </ul> <p>9. Flatten Layer:</p> <ul style="list-style-type: none"> <li>- Flattens the input to prepare for the fully connected layers.</li> </ul> <p>10. Dense Layer 1:</p> <ul style="list-style-type: none"> <li>- Units: 64</li> <li>- Activation: ReLU</li> <li>- Weight Initialization: GlorotUniform</li> <li>- Batch Normalization</li> </ul> <p>11. Dense Layer 2:</p> <ul style="list-style-type: none"> <li>- Units: 32</li> <li>- Activation: ReLU</li> <li>- Weight Initialization: GlorotUniform</li> <li>- Batch Normalization</li> </ul> <p>12. Output Dense Layer:</p> <ul style="list-style-type: none"> <li>- Units: 10 (for 10 classes in classification)</li> <li>- Activation: Softmax</li> <li>- Weight Initialization: GlorotUniform</li> </ul>				
Conf11	<p>1. Convolutional Layer 1:</p> <ul style="list-style-type: none"> <li>- Filters: 32</li> <li>- Kernel Size: (5, 5)</li> <li>- Activation: ReLU</li> <li>- Input Shape: (28, 28, 1)</li> </ul> <p>2. Batch Normalization Layer 1:</p> <ul style="list-style-type: none"> <li>- Applies batch normalization after</li> </ul>	<p>1. Dropout Layer 1:</p> <ul style="list-style-type: none"> <li>- Rate: 0.25</li> <li>- Seed: 42</li> </ul>	SGD momentum 0.9	0.1508	19.7609

	<p>Convolutional Layer 1.</p> <p>3. Max Pooling Layer 1:</p> <ul style="list-style-type: none"> <li>- Pool Size: (3, 3)</li> <li>- Strides: 1</li> </ul> <p>4. Convolutional Layer 2:</p> <ul style="list-style-type: none"> <li>- Filters: 32</li> <li>- Kernel Size: (5, 5)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> <p>5. Batch Normalization Layer 2:</p> <ul style="list-style-type: none"> <li>- Applies batch normalization after Convolutional Layer 2.</li> </ul> <p>6. Max Pooling Layer 2:</p> <ul style="list-style-type: none"> <li>- Pool Size: (3, 3)</li> <li>- Strides: 1</li> </ul> <p>7. Convolutional Layers 3, 4, 5:</p> <ul style="list-style-type: none"> <li>- Convolutional Layer 3:</li> <li>- Filters: 32</li> <li>- Kernel Size: (5, 5)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> <li>- Convolutional Layer 4:</li> <li>- Filters: 32</li> <li>- Kernel Size: (5, 5)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> <li>- Convolutional Layer 5:</li> <li>- Filters: 32</li> <li>- Kernel Size: (5, 5)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> <p>8. Max Pooling Layer 3:</p>				
--	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--	--	--	--

	<ul style="list-style-type: none"> <li>- Pool Size: (3, 3)</li> <li>- Strides: 1</li> </ul> <p>9. Flatten Layer:</p> <ul style="list-style-type: none"> <li>- Flattens the input to prepare for the fully connected layers.</li> </ul> <p>10. Dense Layer 1:</p> <ul style="list-style-type: none"> <li>- Units: 32</li> <li>- Activation: ReLU</li> <li>- Weight Initialization: GlorotUniform</li> </ul> <p>11. Dense Layer 2:</p> <ul style="list-style-type: none"> <li>- Units: 32</li> <li>- Activation: ReLU</li> <li>- Weight Initialization: GlorotUniform</li> <li>- Batch Normalization</li> </ul> <p>12. Output Dense Layer:</p> <ul style="list-style-type: none"> <li>- Units: 10 (for 10 classes in classification)</li> <li>- Activation: Softmax</li> <li>- Weight Initialization: GlorotUniform</li> </ul>				
Conf12	<p>1. Convolutional Layer 1:</p> <ul style="list-style-type: none"> <li>- Filters: 32</li> <li>- Kernel Size: (5, 5)</li> <li>- Activation: ReLU</li> <li>- Input Shape: (28, 28, 1)</li> </ul> <p>2. Batch Normalization Layer 1:</p> <ul style="list-style-type: none"> <li>- Applies batch normalization after Convolutional Layer 1.</li> </ul> <p>3. Max Pooling Layer 1:</p>	<p>1. Dropout Layer 1:</p> <ul style="list-style-type: none"> <li>- Rate: 0.25</li> <li>- Seed: 42</li> </ul>	<p>SGD momentum 0.9</p>	0.6004	0.439

	<ul style="list-style-type: none"> <li>- Pool Size: (3, 3)</li> <li>- Strides: 1</li> </ul> <p>4. Convolutional Layer 2:</p> <ul style="list-style-type: none"> <li>- Filters: 32</li> <li>- Kernel Size: (5, 5)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> <p>5. Batch Normalization Layer 2:</p> <ul style="list-style-type: none"> <li>- Applies batch normalization after Convolutional Layer 2.</li> </ul> <p>6. Max Pooling Layer 2:</p> <ul style="list-style-type: none"> <li>- Pool Size: (3, 3)</li> <li>- Strides: 1</li> </ul> <p>7. Convolutional Layers 3, 4, 5:</p> <ul style="list-style-type: none"> <li>- Convolutional Layer 3:</li> <li>- Filters: 32</li> <li>- Kernel Size: (5, 5)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> <li>- Convolutional Layer 4:</li> <li>- Filters: 32</li> <li>- Kernel Size: (5, 5)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> <li>- Convolutional Layer 5:</li> <li>- Filters: 32</li> <li>- Kernel Size: (5, 5)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> <p>8. Max Pooling Layer 3:</p> <ul style="list-style-type: none"> <li>- Pool Size: (3, 3)</li> <li>- Strides: 1</li> </ul> <p>9. Flatten Layer:</p>				
--	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--	--	--	--

	<ul style="list-style-type: none"> <li>- Flattens the input to prepare for the fully connected layers.</li> </ul> <p>10. Dense Layer 1:</p> <ul style="list-style-type: none"> <li>- Units: 32</li> <li>- Activation: ReLU</li> <li>- Weight Initialization: GlorotUniform</li> </ul> <p>11. Dense Layer 2:</p> <ul style="list-style-type: none"> <li>- Units: 32</li> <li>- Activation: ReLU</li> <li>- Weight Initialization: GlorotUniform</li> <li>- Batch Normalization</li> </ul> <p>12. Output Dense Layer:</p> <ul style="list-style-type: none"> <li>- Units: 10 (for 10 classes in classification)</li> <li>- Activation: Softmax</li> <li>- Weight Initialization: GlorotUniform</li> </ul>				
Conf13	<p>1. Convolutional Layer 1:</p> <ul style="list-style-type: none"> <li>- Filters: 32</li> <li>- Kernel Size: (5, 5)</li> <li>- Activation: ReLU</li> <li>- Input Shape: (28, 28, 1)</li> </ul> <p>2. Batch Normalization Layer 1:</p> <ul style="list-style-type: none"> <li>- Applies batch normalization after Convolutional Layer 1.</li> </ul> <p>3. Max Pooling Layer 1:</p> <ul style="list-style-type: none"> <li>- Pool Size: (3, 3)</li> <li>- Strides: 2</li> </ul> <p>4. Convolutional Layer 2:</p>	<p>1. Dropout Layer 1:</p> <ul style="list-style-type: none"> <li>- Rate: 0.25</li> <li>- Seed: 42</li> </ul> <p>2. Dropout Layer 2:</p> <ul style="list-style-type: none"> <li>- Rate: 0.25</li> <li>- Seed: 42</li> </ul>	SGD momentum 0.9	0.2044	0.2609



	<ul style="list-style-type: none"> <li>- Filters: 32</li> <li>- Kernel Size: (5, 5)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> <p>5. Batch Normalization Layer 2:</p> <ul style="list-style-type: none"> <li>- Applies batch normalization after Convolutional Layer 2.</li> </ul> <p>6. Max Pooling Layer 2:</p> <ul style="list-style-type: none"> <li>- Pool Size: (3, 3)</li> <li>- Strides: 2</li> </ul> <p>7. Convolutional Layers 3, 4, 5:</p> <ul style="list-style-type: none"> <li>- Convolutional Layer 3:</li> <li>- Filters: 32</li> <li>- Kernel Size: (5, 5)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> <li>- Convolutional Layer 4:</li> <li>- Filters: 32</li> <li>- Kernel Size: (5, 5)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> <li>- Convolutional Layer 5:</li> <li>- Filters: 32</li> <li>- Kernel Size: (5, 5)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> <p>8. Max Pooling Layer 3:</p> <ul style="list-style-type: none"> <li>- Pool Size: (3, 3)</li> <li>- Strides: 2</li> </ul> <p>9. Flatten Layer:</p> <ul style="list-style-type: none"> <li>- Flattens the input to prepare for the fully connected layers.</li> </ul>				
--	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--	--	--	--

	<p>10. Dense Layer 1:</p> <ul style="list-style-type: none"> <li>- Units: 32</li> <li>- Activation: ReLU</li> <li>- Weight Initialization: GlorotUniform</li> <li>- Batch Normalization</li> </ul> <p>11. Dense Layer 2:</p> <ul style="list-style-type: none"> <li>- Units: 32</li> <li>- Activation: ReLU</li> <li>- Weight Initialization: GlorotUniform</li> <li>- Batch Normalization</li> </ul> <p>12. Output Dense Layer:</p> <ul style="list-style-type: none"> <li>- Units: 10 (for 10 classes in classification)</li> <li>- Activation: Softmax</li> <li>- Weight Initialization: GlorotUniform</li> </ul>				
Conf14	<p>1. Convolutional Layer 1:</p> <ul style="list-style-type: none"> <li>- Filters: 64</li> <li>- Kernel Size: (4, 4)</li> <li>- Activation: ReLU</li> <li>- Input Shape: (28, 28, 1)</li> </ul> <p>2. Batch Normalization Layer 1:</p> <ul style="list-style-type: none"> <li>- Applies batch normalization after Convolutional Layer 1.</li> </ul> <p>3. Max Pooling Layer 1:</p> <ul style="list-style-type: none"> <li>- Pool Size: (4, 4)</li> <li>- Strides: 2</li> </ul> <p>4. Convolutional Layer 2:</p> <ul style="list-style-type: none"> <li>- Filters: 64</li> <li>- Kernel Size: (4, 4)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> <p>5. Batch Normalization Layer 2:</p> <ul style="list-style-type: none"> <li>- Applies batch normalization after Convolutional Layer 2.</li> </ul>	<p>1. Dropout Layer 1:</p> <ul style="list-style-type: none"> <li>- Rate: 0.25</li> <li>- Seed: 42</li> </ul> <p>2. Dropout Layer 2:</p> <ul style="list-style-type: none"> <li>- Rate: 0.25</li> <li>- Seed: 42</li> </ul>	SGD momentum 0.9	0.1937	0.2473

	<p>6. Max Pooling Layer 2:</p> <ul style="list-style-type: none"> <li>- Pool Size: (4, 4)</li> <li>- Strides: 2</li> </ul> <p>7. Convolutional Layers 3, 4, 5:</p> <ul style="list-style-type: none"> <li>- Convolutional Layer 3:</li> <li>- Filters: 64</li> <li>- Kernel Size: (4, 4)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> <li>- Convolutional Layer 4:</li> <li>- Filters: 64</li> <li>- Kernel Size: (4, 4)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> <li>- Convolutional Layer 5:</li> <li>- Filters: 64</li> <li>- Kernel Size: (4, 4)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> <p>8. Max Pooling Layer 3:</p> <ul style="list-style-type: none"> <li>- Pool Size: (4, 4)</li> <li>- Strides: 2</li> </ul> <p>9. Flatten Layer:</p> <ul style="list-style-type: none"> <li>- Flattens the input to prepare for the fully connected layers.</li> </ul> <p>10. Dense Layer 1:</p> <ul style="list-style-type: none"> <li>- Units: 64</li> <li>- Activation: ReLU</li> <li>- Weight Initialization: GlorotUniform</li> <li>- Batch Normalization</li> </ul> <p>11. Dense Layer 2:</p> <ul style="list-style-type: none"> <li>- Units: 32</li> <li>- Activation: ReLU</li> <li>- Weight Initialization: GlorotUniform</li> <li>- Batch Normalization</li> </ul> <p>12. Output Dense Layer:</p> <ul style="list-style-type: none"> <li>- Units: 10 (for 10 classes in classification)</li> <li>- Activation: Softmax</li> <li>- Weight Initialization: GlorotUniform</li> </ul>				
--	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--	--	--	--

Conf15	<p>1. Convolutional Layer 1:</p> <ul style="list-style-type: none"> <li>- Filters: 32</li> <li>- Kernel Size: (10, 10)</li> <li>- Activation: ReLU</li> <li>- Input Shape: (28, 28, 1)</li> </ul> <p>2. Batch Normalization Layer 1:</p> <ul style="list-style-type: none"> <li>- Applies batch normalization after Convolutional Layer 1.</li> </ul> <p>3. Max Pooling Layer 1:</p> <ul style="list-style-type: none"> <li>- Pool Size: (2, 2)</li> <li>- Strides: 2</li> </ul> <p>4. Convolutional Layer 2:</p> <ul style="list-style-type: none"> <li>- Filters: 64</li> <li>- Kernel Size: (5, 5)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> <p>5. Batch Normalization Layer 2:</p> <ul style="list-style-type: none"> <li>- Applies batch normalization after Convolutional Layer 2.</li> </ul> <p>6. Max Pooling Layer 2:</p> <ul style="list-style-type: none"> <li>- Pool Size: (2, 2)</li> <li>- Strides: 2</li> </ul> <p>7. Convolutional Layers 3, 4, 5:</p> <ul style="list-style-type: none"> <li>- Convolutional Layer 3:</li> <li>- Filters: 64</li> <li>- Kernel Size: (3, 3)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> <li>- Convolutional Layer 4:</li> <li>- Filters: 64</li> <li>- Kernel Size: (3, 3)</li> </ul>	<p>1. Dropout Layer 1:</p> <ul style="list-style-type: none"> <li>- Rate: 0.25</li> <li>- Seed: 42</li> </ul> <p>2. Dropout Layer 2:</p> <ul style="list-style-type: none"> <li>- Rate: 0.25</li> <li>- Seed: 42</li> </ul>	SGD momentum 0.9	0.1774	0.2808
--------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	------------------------	--------	--------

	<ul style="list-style-type: none"> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> <li>- Convolutional Layer 5:</li> <li>- Filters: 32</li> <li>- Kernel Size: (3, 3)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> <p>8. Max Pooling Layer 3:</p> <ul style="list-style-type: none"> <li>- Pool Size: (2, 2)</li> <li>- Strides: 1</li> </ul> <p>9. Flatten Layer:</p> <ul style="list-style-type: none"> <li>- Flattens the input to prepare for the fully connected layers.</li> </ul> <p>10. Dense Layer 1:</p> <ul style="list-style-type: none"> <li>- Units: 128</li> <li>- Activation: ReLU</li> <li>- Weight Initialization: GlorotUniform</li> <li>- Batch Normalization</li> </ul> <p>11. Dense Layer 2:</p> <ul style="list-style-type: none"> <li>- Units: 64</li> <li>- Activation: ReLU</li> <li>- Weight Initialization: GlorotUniform</li> <li>- Batch Normalization</li> </ul> <p>12. Output Dense Layer:</p> <ul style="list-style-type: none"> <li>- Units: 20 (for 20 classes in classification)</li> <li>- Activation: Softmax</li> <li>- Weight Initialization: GlorotUniform</li> </ul>				
Conf16	<p>1. Convolutional Layer 1:</p> <ul style="list-style-type: none"> <li>- Filters: 32</li> <li>- Kernel Size: (7, 7)</li> </ul>	<p>1. Dropout Layer 1:</p> <ul style="list-style-type: none"> <li>- Rate: 0.5</li> </ul>	SGD momentum 0.9	0.3023	0.2944

	<ul style="list-style-type: none"> <li>- Activation: ReLU</li> <li>- Input Shape: (28, 28, 1)</li> </ul> <p>2. Batch Normalization Layer 1:</p> <ul style="list-style-type: none"> <li>- Applies batch normalization after Convolutional Layer 1.</li> </ul> <p>3. Max Pooling Layer 1:</p> <ul style="list-style-type: none"> <li>- Pool Size: (3, 3)</li> <li>- Strides: 1</li> </ul> <p>4. Convolutional Layer 2:</p> <ul style="list-style-type: none"> <li>- Filters: 64</li> <li>- Kernel Size: (2, 2)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> <p>5. Batch Normalization Layer 2:</p> <ul style="list-style-type: none"> <li>- Applies batch normalization after Convolutional Layer 2.</li> </ul> <p>6. Max Pooling Layer 2:</p> <ul style="list-style-type: none"> <li>- Pool Size: (3, 3)</li> <li>- Strides: 2</li> </ul> <p>7. Convolutional Layers 3, 4, 5:</p> <ul style="list-style-type: none"> <li>- Convolutional Layer 3:</li> <li>- Filters: 64</li> <li>- Kernel Size: (5, 5)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> <li>- Convolutional Layer 4:</li> <li>- Filters: 64</li> <li>- Kernel Size: (4, 4)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> <li>- Convolutional Layer 5:</li> </ul>	<ul style="list-style-type: none"> <li>- Seed: 42</li> </ul> <p>2. Dropout Layer 2:</p> <ul style="list-style-type: none"> <li>- Rate: 0.5</li> <li>- Seed: 42</li> </ul>			
--	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--	--	--

	<ul style="list-style-type: none"> <li>- Filters: 32</li> <li>- Kernel Size: (10, 10)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> <p>8. Max Pooling Layer 3:</p> <ul style="list-style-type: none"> <li>- Pool Size: (3, 3)</li> <li>- Strides: 1</li> </ul> <p>9. Flatten Layer:</p> <ul style="list-style-type: none"> <li>- Flattens the input to prepare for the fully connected layers.</li> </ul> <p>10. Dense Layer 1:</p> <ul style="list-style-type: none"> <li>- Units: 64</li> <li>- Activation: ReLU</li> <li>- Weight Initialization: GlorotUniform</li> <li>- Batch Normalization</li> </ul> <p>11. Dense Layer 2:</p> <ul style="list-style-type: none"> <li>- Units: 32</li> <li>- Activation: ReLU</li> <li>- Weight Initialization: GlorotUniform</li> <li>- Batch Normalization</li> </ul> <p>12. Output Dense Layer:</p> <ul style="list-style-type: none"> <li>- Units: 15 (for 15 classes in classification)</li> <li>- Activation: Softmax</li> <li>- Weight Initialization: GlorotUniform</li> </ul>				
Conf17	<p>1. Convolutional Layer 1:</p> <ul style="list-style-type: none"> <li>- Filters: 64</li> <li>- Kernel Size: (7, 7)</li> <li>- Activation: ReLU</li> <li>- Input Shape: (28, 28, 1)</li> </ul>	<p>1. Dropout Layer 1:</p> <ul style="list-style-type: none"> <li>- Rate: 0.5</li> <li>- Seed: 42</li> </ul> <p>2. Dropout</p>	SGD momentum 0.9	0.2623	0.2549

	<p>2. Batch Normalization Layer 1:</p> <ul style="list-style-type: none"> <li>- Applies batch normalization after Convolutional Layer 1.</li> </ul> <p>3. Max Pooling Layer 1:</p> <ul style="list-style-type: none"> <li>- Pool Size: (3, 3)</li> <li>- Strides: 1</li> </ul> <p>4. Convolutional Layer 2:</p> <ul style="list-style-type: none"> <li>- Filters: 64</li> <li>- Kernel Size: (8, 8)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> <p>5. Batch Normalization Layer 2:</p> <ul style="list-style-type: none"> <li>- Applies batch normalization after Convolutional Layer 2.</li> </ul> <p>6. Max Pooling Layer 2:</p> <ul style="list-style-type: none"> <li>- Pool Size: (3, 3)</li> <li>- Strides: 2</li> </ul> <p>7. Convolutional Layers 3, 4, 5:</p> <ul style="list-style-type: none"> <li>- Convolutional Layer 3:</li> <li>- Filters: 64</li> <li>- Kernel Size: (5, 5)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> <li>- Convolutional Layer 4:</li> <li>- Filters: 64</li> <li>- Kernel Size: (4, 4)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> <li>- Convolutional Layer 5:</li> <li>- Filters: 64</li> <li>- Kernel Size: (10, 10)</li> <li>- Activation: ReLU</li> </ul>	<p>Layer 2:</p> <ul style="list-style-type: none"> <li>- Rate: 0.5</li> <li>- Seed: 42</li> </ul>			
--	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------	--	--	--



	<ul style="list-style-type: none"> <li>- Padding: 'same'</li> </ul> <p>8. Max Pooling Layer 3:</p> <ul style="list-style-type: none"> <li>- Pool Size: (3, 3)</li> <li>- Strides: 1</li> </ul> <p>9. Flatten Layer:</p> <ul style="list-style-type: none"> <li>- Flattens the input to prepare for the fully connected layers.</li> </ul> <p>10. Dense Layer 1:</p> <ul style="list-style-type: none"> <li>- Units: 128</li> <li>- Activation: ReLU</li> <li>- Weight Initialization: GlorotUniform</li> <li>- Batch Normalization</li> </ul> <p>11. Dense Layer 2:</p> <ul style="list-style-type: none"> <li>- Units: 64</li> <li>- Activation: ReLU</li> <li>- Weight Initialization: GlorotUniform</li> <li>- Batch Normalization</li> </ul> <p>12. Output Dense Layer:</p> <ul style="list-style-type: none"> <li>- Units: 15 (for 15 classes in classification)</li> <li>- Activation: Softmax</li> <li>- Weight Initialization: GlorotUniform</li> </ul>				
Conf18	<p>1. Convolutional Layer 1:</p> <ul style="list-style-type: none"> <li>- Filters: 64</li> <li>- Kernel Size: (6, 6)</li> <li>- Activation: ReLU</li> <li>- Input Shape: (28, 28, 1)</li> </ul> <p>2. Batch Normalization Layer 1:</p> <ul style="list-style-type: none"> <li>- Applies batch</li> </ul>	<p>1. Dropout Layer 1:</p> <ul style="list-style-type: none"> <li>- Rate: 0.05</li> <li>- Seed: 42</li> </ul> <p>2. Dropout Layer 2:</p> <ul style="list-style-type: none"> <li>- Rate: 0.05</li> <li>- Seed: 42</li> </ul>	SGD momentum 0.9	0.1505	0.2499

	<p>normalization after Convolutional Layer 1.</p> <p>3. Max Pooling Layer 1:</p> <ul style="list-style-type: none"> <li>- Pool Size: (3, 3)</li> <li>- Strides: 1</li> </ul> <p>4. Convolutional Layer 2:</p> <ul style="list-style-type: none"> <li>- Filters: 64</li> <li>- Kernel Size: (5, 5)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> <p>5. Batch Normalization Layer 2:</p> <ul style="list-style-type: none"> <li>- Applies batch normalization after Convolutional Layer 2.</li> </ul> <p>6. Max Pooling Layer 2:</p> <ul style="list-style-type: none"> <li>- Pool Size: (3, 3)</li> <li>- Strides: 1</li> </ul> <p>7. Convolutional Layers 3, 4, 5:</p> <ul style="list-style-type: none"> <li>- Convolutional Layer 3:</li> <li>- Filters: 64</li> <li>- Kernel Size: (4, 4)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> <li>- Convolutional Layer 4:</li> <li>- Filters: 64</li> <li>- Kernel Size: (3, 3)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> <li>- Convolutional Layer 5:</li> <li>- Filters: 64</li> <li>- Kernel Size: (2, 2)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> <p>8. Max Pooling Layer 3:</p>				
--	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--	--	--	--

	<ul style="list-style-type: none"> <li>- Pool Size: (3, 3)</li> <li>- Strides: 1</li> </ul> <p>9. Flatten Layer:</p> <ul style="list-style-type: none"> <li>- Flattens the input to prepare for the fully connected layers.</li> </ul> <p>10. Dense Layer 1:</p> <ul style="list-style-type: none"> <li>- Units: 64</li> <li>- Activation: ReLU</li> <li>- Weight Initialization: GlorotUniform</li> <li>- Batch Normalization</li> </ul> <p>11. Dense Layer 2:</p> <ul style="list-style-type: none"> <li>- Units: 64</li> <li>- Activation: ReLU</li> <li>- Weight Initialization: GlorotUniform</li> <li>- Batch Normalization</li> </ul> <p>12. Output Dense Layer:</p> <ul style="list-style-type: none"> <li>- Units: 15 (for 15 classes in classification)</li> <li>- Activation: Softmax</li> <li>- Weight Initialization: GlorotUniform</li> </ul>				
Conf19	<p>1. Convolutional Layer 1:</p> <ul style="list-style-type: none"> <li>- Filters: 64</li> <li>- Kernel Size: (8, 8)</li> <li>- Activation: ReLU</li> <li>- Input Shape: (28, 28, 1)</li> </ul> <p>2. Batch Normalization Layer 1:</p> <ul style="list-style-type: none"> <li>- Applies batch normalization after Convolutional Layer 1.</li> </ul>	<p>1. Dropout Layer 1:</p> <ul style="list-style-type: none"> <li>- Rate: 0.15</li> <li>- Seed: 42</li> </ul> <p>2. Dropout Layer 2:</p> <ul style="list-style-type: none"> <li>- Rate: 0.15</li> <li>- Seed: 42</li> </ul>	SGD momentum 0.9	0.1476	0.2496

	<p>3. Max Pooling Layer 1:</p> <ul style="list-style-type: none"> <li>- Pool Size: (3, 3)</li> <li>- Strides: 2</li> </ul> <p>4. Convolutional Layer 2:</p> <ul style="list-style-type: none"> <li>- Filters: 64</li> <li>- Kernel Size: (5, 5)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> <p>5. Batch Normalization Layer 2:</p> <ul style="list-style-type: none"> <li>- Applies batch normalization after Convolutional Layer 2.</li> </ul> <p>6. Max Pooling Layer 2:</p> <ul style="list-style-type: none"> <li>- Pool Size: (3, 3)</li> <li>- Strides: 2</li> </ul> <p>7. Convolutional Layers 3, 4, 5:</p> <ul style="list-style-type: none"> <li>- Convolutional Layer 3:</li> <li>- Filters: 64</li> <li>- Kernel Size: (4, 4)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> <li>- Convolutional Layer 4:</li> <li>- Filters: 64</li> <li>- Kernel Size: (4, 4)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> <li>- Convolutional Layer 5:</li> <li>- Filters: 64</li> <li>- Kernel Size: (4, 4)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> <p>8. Max Pooling Layer 3:</p> <ul style="list-style-type: none"> <li>- Pool Size: (3, 3)</li> <li>- Strides: 2</li> </ul>				
--	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--	--	--	--

	<p>9. Flatten Layer:</p> <ul style="list-style-type: none"> <li>- Flattens the input to prepare for the fully connected layers.</li> </ul> <p>10. Dense Layer 1:</p> <ul style="list-style-type: none"> <li>- Units: 128</li> <li>- Activation: ReLU</li> <li>- Weight Initialization: GlorotUniform</li> <li>- Batch Normalization</li> </ul> <p>11. Dense Layer 2:</p> <ul style="list-style-type: none"> <li>- Units: 64</li> <li>- Activation: ReLU</li> <li>- Weight Initialization: GlorotUniform</li> <li>- Batch Normalization</li> </ul> <p>12. Output Dense Layer:</p> <ul style="list-style-type: none"> <li>- Units: 25 (for 25 classes in classification)</li> <li>- Activation: Softmax</li> <li>- Weight Initialization: GlorotUniform</li> </ul>				
Conf20	<p>1. Convolutional Layer 1:</p> <ul style="list-style-type: none"> <li>- Filters: 32</li> <li>- Kernel Size: (6, 6)</li> <li>- Activation: ReLU</li> <li>- Input Shape: (28, 28, 1)</li> </ul> <p>2. Batch Normalization Layer 1:</p> <ul style="list-style-type: none"> <li>- Applies batch normalization after Convolutional Layer 1.</li> </ul> <p>3. Max Pooling Layer 1:</p> <ul style="list-style-type: none"> <li>- Pool Size: (3, 3)</li> <li>- Strides: 1</li> </ul>	<p>1. Dropout Layer 1:</p> <ul style="list-style-type: none"> <li>- Rate: 0.25</li> <li>- Seed: 42</li> </ul> <p>2. Dropout Layer 2:</p> <ul style="list-style-type: none"> <li>- Rate: 0.25</li> <li>- Seed: 42</li> </ul>	SGD momentum 0.9	0.2094	0.2404

	<p>4. Convolutional Layer 2:</p> <ul style="list-style-type: none"> <li>- Filters: 64</li> <li>- Kernel Size: (6, 6)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> <p>5. Batch Normalization Layer 2:</p> <ul style="list-style-type: none"> <li>- Applies batch normalization after Convolutional Layer 2.</li> </ul> <p>6. Max Pooling Layer 2:</p> <ul style="list-style-type: none"> <li>- Pool Size: (3, 3)</li> <li>- Strides: 2</li> </ul> <p>7. Convolutional Layers 3, 4, 5:</p> <ul style="list-style-type: none"> <li>- Convolutional Layer 3: <ul style="list-style-type: none"> <li>- Filters: 64</li> <li>- Kernel Size: (4, 4)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> </li> <li>- Convolutional Layer 4: <ul style="list-style-type: none"> <li>- Filters: 64</li> <li>- Kernel Size: (4, 4)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> </li> <li>- Convolutional Layer 5: <ul style="list-style-type: none"> <li>- Filters: 32</li> <li>- Kernel Size: (4, 4)</li> <li>- Activation: ReLU</li> <li>- Padding: 'same'</li> </ul> </li> </ul> <p>8. Max Pooling Layer 3:</p> <ul style="list-style-type: none"> <li>- Pool Size: (3, 3)</li> <li>- Strides: 1</li> </ul> <p>9. Flatten Layer:</p> <ul style="list-style-type: none"> <li>- Flattens the input to prepare for the fully</li> </ul>				
--	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--	--	--	--

	connected layers.				
	10. Dense Layer 1: - Units: 64 - Activation: ReLU - Weight Initialization: GlorotUniform - Batch Normalization				
	11. Dense Layer 2: - Units: 32 - Activation: ReLU - Weight Initialization: GlorotUniform - Batch Normalization				
	12. Output Dense Layer: - Units: 10 (for 10 classes in classification) - Activation: Softmax - Weight Initialization: GlorotUniform				

## Analisis

Pendekatan menggunakan arsitektur yang lebih kompleks seperti AlexNet pada dataset Fashion-MNIST yang relatif sederhana menghasilkan model yang kurang bagus, karena dataset Fashion-MNIST adalah gambar hitam-putih dan ukurannya lebih kecil, tidak seperti CIFAR yang memiliki gambar berwarna dan ukuran lebih besar.


Awalnya kami mencoba menggunakan hyperparameter default dari AlexNet kami mendapatkan accuracy yang lumayan bagus namun loss nya masih tinggi / tidak stabil. Kemudian kami mencoba beberapa konfigurasi dan menemukan bahwa semakin simple modelnya semakin baik accuracy dan lossnya.

Jadi, dalam konteks Fashion-MNIST penggunaan model yang lebih sederhana atau arsitektur yang lebih ringan seringkali lebih cocok dan dapat memberikan kinerja yang baik tanpa mengorbankan waktu training atau computational cost yang tinggi.

# Reference

[Convolutional Neural Network \(CNN\) | TensorFlow Core](#)

[Conv2D layer](#)

 Image classification using CNN (CIFAR10 dataset) | Deep Learning Tutorial 24 (Tensorflo...