מטלה 2

מגיש: אור שני 206812034

ניסוי ראשון

תוצאה:

Epoch 36/50

1875/1875 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ **5s** 3ms/step - loss: 0.1444 - sparse_categorical_accuracy: 0.9707 - val_loss: 0.1299 - val_sparse_categorical_accuracy: 0.9774

שכבות:

```
layers = [
  tf.keras.layers.Flatten(input_shape=image_shape),

  tf.keras.layers.Dense(32, kernel_regularizer=tf.keras.regularizers.l2(0.0001)),
  tf.keras.layers.BatchNormalization(),
  tf.keras.layers.Activation('relu'),


  tf.keras.layers.Dense(64, kernel_regularizer=tf.keras.regularizers.l2(0.001)),
  tf.keras.layers.BatchNormalization(),
  tf.keras.layers.Activation('relu'),

  tf.keras.layers.Dense(128, kernel_regularizer=tf.keras.regularizers.l2(0.01)),
  tf.keras.layers.BatchNormalization(),
  tf.keras.layers.Activation('relu'),

  tf.keras.layers.Dense(num_of_classes),
  tf.keras.layers.Softmax()
  ]
```

ניסוי שני

תוצאה:

Epoch 25/50

1875/1875 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 6s 3ms/step - loss: 0.1757 - sparse_categorical_accuracy: 0.9608 - val_loss: 0.1589 - val_sparse_categorical_accuracy: 0.9700

שכבות:

```
layers = [
  tf.keras.layers.Flatten(input_shape=image_shape),

  tf.keras.layers.Dense(32, kernel_regularizer=tf.keras.regularizers.l2(0.0002)),
  tf.keras.layers.BatchNormalization(),
  tf.keras.layers.Activation('sigmoid'),


  tf.keras.layers.Dense(64, kernel_regularizer=tf.keras.regularizers.l2(0.002)),
  tf.keras.layers.BatchNormalization(),
  tf.keras.layers.Activation('sigmoid'),

  tf.keras.layers.Dense(128, kernel_regularizer=tf.keras.regularizers.l2(0.02)),
  tf.keras.layers.BatchNormalization(),
  tf.keras.layers.Activation('sigmoid'),

  tf.keras.layers.Dense(num_of_classes),
  tf.keras.layers.Softmax()
  ]
```

ניסוי שלישי

תוצאה:

Epoch 50/50

1875/1875 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ **10s** 3ms/step - loss: 0.2533 - sparse_categorical_accuracy: 0.9413 - val_loss: 0.1550 - val_sparse_categorical_accuracy: 0.9711

שכבות:

```
layers = [
  tf.keras.layers.Flatten(input_shape=image_shape),

  tf.keras.layers.Dense(32, kernel_regularizer=tf.keras.regularizers.l2(0.0001)),
  tf.keras.layers.BatchNormalization(),
  tf.keras.layers.Activation('relu'),
  tf.keras.layers.Dropout(0.1),

  tf.keras.layers.Dense(64, kernel_regularizer=tf.keras.regularizers.l2(0.001)),
  tf.keras.layers.BatchNormalization(),
  tf.keras.layers.Activation('relu'),
  tf.keras.layers.Dropout(0.1),

  tf.keras.layers.Dense(128, kernel_regularizer=tf.keras.regularizers.l2(0.01)),
  tf.keras.layers.BatchNormalization(),
  tf.keras.layers.Activation('relu'),
  tf.keras.layers.Dropout(0.1),

  tf.keras.layers.Dense(num_of_classes),
  tf.keras.layers.Softmax()
  ]
```

ניסוי רביעי

תוצאה:

Epoch 24/50

1875/1875 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ **9s** 3ms/step - loss: 0.2837 - sparse_categorical_accuracy: 0.9318 - val_loss: 0.1671 - val_sparse_categorical_accuracy: 0.9690

שכבות:

```
layers = [
  tf.keras.layers.Flatten(input_shape=image_shape),

  tf.keras.layers.Dense(32, kernel_regularizer=tf.keras.regularizers.l2(0.0001)),
  tf.keras.layers.BatchNormalization(),
  tf.keras.layers.Activation('sigmoid'),
  tf.keras.layers.Dropout(0.1),

  tf.keras.layers.Dense(64, kernel_regularizer=tf.keras.regularizers.l2(0.001)),
  tf.keras.layers.BatchNormalization(),
  tf.keras.layers.Activation('sigmoid'),
  tf.keras.layers.Dropout(0.1),

  tf.keras.layers.Dense(128, kernel_regularizer=tf.keras.regularizers.l2(0.01)),
  tf.keras.layers.BatchNormalization(),
  tf.keras.layers.Activation('sigmoid'),
  tf.keras.layers.Dropout(0.1),

  tf.keras.layers.Dense(num_of_classes),
  tf.keras.layers.Softmax()
  ]
```

ניסוי חמישי

תוצאה:

Epoch 46/50

1875/1875 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ **5s** 2ms/step - loss: 0.0391 - sparse_categorical_accuracy: 0.9914 - val_loss: 0.3822 - val_sparse_categorical_accuracy: 0.9748

שכבות:

```
layers = [
  tf.keras.layers.Flatten(input_shape=image_shape),

  tf.keras.layers.Dense(32),
  tf.keras.layers.Activation('relu'),

  tf.keras.layers.Dense(64),
  tf.keras.layers.Activation('relu'),

  tf.keras.layers.Dense(128),
  tf.keras.layers.Activation('relu'),

  tf.keras.layers.Dense(num_of_classes),
  tf.keras.layers.Softmax()
  ]
```

ניסוי שישי

תוצאה:

Epoch 39/50

1875/1875 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ **10s** 3ms/step - loss: 0.0372 - sparse_categorical_accuracy: 0.9875 - val_loss: 0.0884 - val_sparse_categorical_accuracy: 0.978

שכבות:

```
layers = [
  tf.keras.layers.Flatten(input_shape=image_shape),

  tf.keras.layers.Dense(32),
  tf.keras.layers.BatchNormalization(),
  tf.keras.layers.Activation('sigmoid'),

  tf.keras.layers.Dense(64),
  tf.keras.layers.BatchNormalization(),
  tf.keras.layers.Activation('sigmoid'),

  tf.keras.layers.Dense(128),
  tf.keras.layers.BatchNormalization(),
  tf.keras.layers.Activation('sigmoid'),

  tf.keras.layers.Dense(num_of_classes),
  tf.keras.layers.Softmax()
  ]
```

<u>ניסוי שביעי</u>

תוצאה:

Epoch 39/50

1875/1875 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ **6s** 3ms/step - loss: 0.0309 - sparse_categorical_accuracy: 0.9901 - val_loss: 0.0832 - val_sparse_categorical_accuracy: 0.9790

שכבות:

```
layers = [
  tf.keras.layers.Flatten(input_shape=image_shape),

  tf.keras.layers.Dense(32),
  tf.keras.layers.BatchNormalization(),
  tf.keras.layers.Activation('tanh'),

  tf.keras.layers.Dense(64),
  tf.keras.layers.BatchNormalization(),
  tf.keras.layers.Activation('tanh'),

  tf.keras.layers.Dense(128),
  tf.keras.layers.BatchNormalization(),
  tf.keras.layers.Activation('tanh'),

  tf.keras.layers.Dense(num_of_classes),
  tf.keras.layers.Softmax()
  ]
```

ניסוי שמיני

תוצאה:

Epoch 36/50

1875/1875 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 5s 2ms/step - loss: 0.1460 - sparse_categorical_accuracy: 0.9692 - val_loss: 0.1438 - val_sparse_categorical_accuracy: 0.9717

שכבות:

```
layers = [
  tf.keras.layers.Flatten(input_shape=image_shape),

  tf.keras.layers.Dense(32, kernel_regularizer=tf.keras.regularizers.l2(0.0001)),
  tf.keras.layers.BatchNormalization(),
  tf.keras.layers.Activation('tanh'),

  tf.keras.layers.Dense(64, kernel_regularizer=tf.keras.regularizers.l2(0.001)),
  tf.keras.layers.BatchNormalization(),
  tf.keras.layers.Activation('tanh'),

  tf.keras.layers.Dense(128, kernel_regularizer=tf.keras.regularizers.l2(0.01)),
  tf.keras.layers.BatchNormalization(),
  tf.keras.layers.Activation('tanh'),

  tf.keras.layers.Dense(num_of_classes),
  tf.keras.layers.Softmax()
  ]
```

ניסוי תשיעי

תוצאה:

Epoch 27/50

1875/1875 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 3s 2ms/step - loss: 0.0141 - sparse_categorical_accuracy: 0.9952 - val_loss: 0.1505 - val_sparse_categorical_accuracy: 0.9717

שכבות:

```
layers = [
  tf.keras.layers.Flatten(input_shape=image_shape),

  tf.keras.layers.Dense(32),
  tf.keras.layers.Activation('tanh'),

  tf.keras.layers.Dense(64),
  tf.keras.layers.Activation('tanh'),

  tf.keras.layers.Dense(128),
  tf.keras.layers.Activation('tanh'),

  tf.keras.layers.Dense(num_of_classes),
  tf.keras.layers.Softmax()
  ]
```

ניסוי עשירי

תוצאה:

Epoch 48/50

1875/1875 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ **6s** 3ms/step - loss: 0.1480 - sparse_categorical_accuracy: 0.9544 - val_loss: 0.0936 - val_sparse_categorical_accuracy: 0.9735

שכבות:

```
layers = [
  tf.keras.layers.Flatten(input_shape=image_shape),

  tf.keras.layers.Dense(32),
  tf.keras.layers.BatchNormalization(),
  tf.keras.layers.Activation('tanh'),
  tf.keras.layers.Dropout(0.1),

  tf.keras.layers.Dense(64),
  tf.keras.layers.BatchNormalization(),
  tf.keras.layers.Activation('tanh'),
  tf.keras.layers.Dropout(0.2),

  tf.keras.layers.Dense(128),
  tf.keras.layers.BatchNormalization(),
  tf.keras.layers.Activation('tanh'),
  tf.keras.layers.Dropout(0.3),

  tf.keras.layers.Dense(num_of_classes),
  tf.keras.layers.Softmax()
  ]
```

ניסוי 11

תוצאה:

Epoch 39/50

1875/1875 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ **6s** 3ms/step - loss: 0.1265 - sparse_categorical_accuracy: 0.9723 - val_loss: 0.1279 - val_sparse_categorical _accuracy: 0.9737

שכבות:

```
layers = [
  tf.keras.layers.Flatten(input_shape=image_shape),

  tf.keras.layers.Dense(32, kernel_regularizer=tf.keras.regularizers.l2(0.0001)),
  tf.keras.layers.BatchNormalization(),
  tf.keras.layers.Activation('relu'),


  tf.keras.layers.Dense(64, kernel_regularizer=tf.keras.regularizers.l2(0.001)),
  tf.keras.layers.BatchNormalization(),
  tf.keras.layers.Activation('sigmoid'),

  tf.keras.layers.Dense(128, kernel_regularizer=tf.keras.regularizers.l2(0.01)),
  tf.keras.layers.BatchNormalization(),
  tf.keras.layers.Activation('tanh'),

  tf.keras.layers.Dense(num_of_classes),
  tf.keras.layers.Softmax()
  ]
```

ניסוי 12

תוצאה:

Epoch 20/50

1875/1875 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ **9s** 2ms/step - loss: 0.0600 - sparse_categorical_accuracy: 0.9807 - val_loss: 0.0793 - val_sparse_categorical_accuracy: 0.9777

שכבות:

```
layers = [
  tf.keras.layers.Flatten(input_shape=image_shape),

  tf.keras.layers.Dense(32),
  tf.keras.layers.BatchNormalization(),
  tf.keras.layers.Activation('relu'),


  tf.keras.layers.Dense(64),
  tf.keras.layers.BatchNormalization(),
  tf.keras.layers.Activation('sigmoid'),

  tf.keras.layers.Dense(128),
  tf.keras.layers.BatchNormalization(),
  tf.keras.layers.Activation('tanh'),

  tf.keras.layers.Dense(num_of_classes),
  tf.keras.layers.Softmax()
  ]
```

ניסוי 13

תוצאה:

Epoch 13/50

1875/1875 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ **4s** 2ms/step - loss: 0.0493 - sparse_categorical_accuracy: 0.9842 - val_loss: 0.1064 - val_sparse_categorical_accuracy: 0.9724

שכבות:

```
layers = [
  tf.keras.layers.Flatten(input_shape=image_shape),

  tf.keras.layers.Dense(32),
  tf.keras.layers.Activation('relu'),


  tf.keras.layers.Dense(64),
  tf.keras.layers.Activation('sigmoid'),

  tf.keras.layers.Dense(128),
  tf.keras.layers.Activation('tanh'),

  tf.keras.layers.Dense(num_of_classes),
  tf.keras.layers.Softmax()
  ]
```

ניסוי 14

תוצאה:

Epoch 50/50

1875/1875 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ **4s** 2ms/step - loss: 0.1093 - sparse_categorical_accuracy: 0.9696 - val_loss: 0.1240 - val_sparse_categorical_accuracy: 0.9704

שכבות:

```
layers = [
  tf.keras.layers.Flatten(input_shape=image_shape),

  tf.keras.layers.Dense(32),
  tf.keras.layers.Activation('relu'),
  tf.keras.layers.Dropout(0.1),

  tf.keras.layers.Dense(64),
  tf.keras.layers.Activation('sigmoid'),
  tf.keras.layers.Dropout(0.2),

  tf.keras.layers.Dense(128),
  tf.keras.layers.Activation('tanh'),
  tf.keras.layers.Dropout(0.3),

  tf.keras.layers.Dense(num_of_classes),
  tf.keras.layers.Softmax()
  ]
```

ניסוי 15

תוצאה:

Epoch 35/50

1875/1875 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ **5s** 3ms/step - loss: 0.1678 - sparse_categorical_accuracy: 0.9686 - val_loss: 0.1740 - val_sparse_categorical_accuracy: 0.9708

שכבות:

```
layers = [
  tf.keras.layers.Flatten(input_shape=image_shape),

  tf.keras.layers.Dense(32, kernel_regularizer=tf.keras.regularizers.l2(0.0001)),
  tf.keras.layers.BatchNormalization(),
  tf.keras.layers.Activation('sigmoid'),


  tf.keras.layers.Dense(64, kernel_regularizer=tf.keras.regularizers.l2(0.001)),
  tf.keras.layers.BatchNormalization(),
  tf.keras.layers.Activation('tanh'),

  tf.keras.layers.Dense(128, kernel_regularizer=tf.keras.regularizers.l2(0.01)),
  tf.keras.layers.BatchNormalization(),
  tf.keras.layers.Activation('relu'),

  tf.keras.layers.Dense(num_of_classes),
  tf.keras.layers.Softmax()
  ]
```