

```
from tensorflow.keras.models import Model
from tensorflow.keras.layers import Input, Dense, Flatten, BatchNormalization

from custom_layers import pipe_model, hidden_stack, conv_stack_2d

def cnn(input_size, exp_cfg):
    """Construct a simple convolutional neural network"""

    layers = []

    # Input batch normalization
    layers.append(
        BatchNormalization(axis=exp_cfg.model.input_axis_norm)
    )

    # Add conv layers with respective l2 values
    layers.append(
        conv_stack_2d(
            filters=exp_cfg.model.conv.filters,
            kernels=exp_cfg.model.conv.kernels,
            strides=exp_cfg.model.conv.strides,
            max_pool_sizes=exp_cfg.model.conv.max_pool_sizes,
            batch_norms=exp_cfg.model.conv.batch_norms,
            l2=exp_cfg.model.conv.l2
        )
    )

    # Flatten for dnn
    layers.append(
        Flatten()
    )

    # Add dnn
    layers.append(
        hidden_stack(
            hidden_sizes=exp_cfg.model.dense.hidden_sizes,
            batch_norms=exp_cfg.model.dense.batch_norms,
            dropout=exp_cfg.model.dense.dropout
        )
    )

    # Add output layer
    layers.append(
        Dense(
            exp_cfg.model.output.output_size,
            activation=exp_cfg.model.output.activation
        )
    )

    # Pipe model by feeding through input placeholder
    inputs = Input(shape=input_size)
    outputs = pipe_model(inputs, layers)

    return Model(inputs=inputs, outputs=outputs)
```