```
from tensorflow.keras.layers import Dense, Dropout, Conv2D, MaxPool2D, BatchNormalization
from tensorflow.keras import regularizers

from .util import pipe_model

def conv_stack_2d(filters, kernels, strides, max_pool_sizes, batch_norms=0, padding="same
", activation="elu", l2=0):
    """Represents a stack of convolutional layers"""

    # If padding is one word or default, extend into a uniform list
    if type(batch_norms) != list:
        batch_norms = [batch_norms]*len(filters)

    # If padding is one word or default, extend into a uniform list
    if type(padding) != list:
        padding = [padding]*len(filters)

    # If activation is one word or default, extend into a uniform list
    if type(activation) != list:
        activation = [activation]*len(filters)

    # Add convolutions
    layers = []
    for i in range(len(filters)):

        if l2 > 0:
            layers.append(Conv2D(
                            filters=filters[i],
                            kernel_size=kernels[i],
                            strides=strides[i],
                            padding=padding[i],
                            activation=activation[i],
                            kernel_regularizer=regularizers.l2(l2)
                ))
        else:
            layers.append(Conv2D(
                            filters=filters[i],
                            kernel_size=kernels[i],
                            strides=strides[i],
                            padding=padding[i],
                            activation=activation[i]
                ))

        if batch_norms[i] == 1:
            layers.append(BatchNormalization(axis=3))

        layers.append(MaxPool2D(pool_size=max_pool_sizes[i]))

    def conv_stack_2d_layer(inputs):
        """Layer hook for stack"""

        return pipe_model(inputs, layers)

    return conv_stack_2d_layer
```