$_model()->$
$Sequential:$
$Definemodel.model=$
$Sequential()model.add(ZeroPadding2D(padding=$
$pad,input_shape=$
$(input_size[0],input_size[1],1),name=$
$"padding_layer"))model.add(Conv2D(conv_filter_num,conv_kernel_size,activation=$
$"relu",padding=$
$"valid",kernel_initializer=$
$"he_uniform",input_shape=$
$(30,30,1),name=$
$"convolution_layer"))model.add(MaxPooling2D(pool_size,name=$
$"max_pooling_layer"))model.add(Flatten(name=$
$"flatten_layer"))model.add(Dense(10,activation=$
$"softmax",name=$
$"dense_layer"))Compilemodel.model.Compile(optimizer=$
$Adam(),loss=$
$"categorical_crossentropy",metrics=$
$["accuracy"])Returnmodel.returnmodel$
conv_weights.h
dense_weights.h
definitions.h
conv_weights.h
dense_weights.h
definitions.h
in.dat
out.dat
gen_data.ipynb
$_to_dense_stream,intfilter,hls::$
$stream<$
$float>$
$dense_to_softmax_stream)floatflat_value;floatdense_array[DENSE_SIZE]=0;$
$_for_flat:$
$for(inti=$
$0;i<$
$FLAT_SIZE/FILTERS;++$
$i)flat_value=flat_to_dense_stream.read();$
$_SIZE;++$
$d)intindex=filter*(FLAT_SIZE/FILTERS)+i;dense_array[d]+=dense_weights[index][d]*flat_value;$
$_SIZE;++$
$j)dense_to_softmax_stream.write(dense_array[j]);$
$_layer(floatpad_img0[PAD_IMG_ROWS][PAD_IMG_COLS],floatpad_img1[PAD_IMG_ROWS][PAD_IMG_COLS],floatpad$
$stream<$
$float>$
$conv_to_pool_streams[FILTERS])convolution(pad_img0,0,conv_to_pool_streams[0]);convolution(pad_img1,1,conv_to_pool_strea$
flattening
$_to_flat_stream,hls::$
$stream<$
$float>$
$flat_to_dense_stream)flat_for_rows:for(intr=0;r<POOL_IMG_ROWS;++r)flat_for_cols:for(intc=0;c<POOL_IM$
CNN
$_in[IMG_ROWS][IMG_COLS],floatprediction[DIGITS])/********Pre-processingdata.********/$
$_img0[PAD_IMG_ROWS][PAD_IMG_COLS]=$
$0;normalization_and_padding(img_in,pad_img0);$
$_SYNTHESIS_{printf("Paddedimage.");print_pad_img(pad_img);endifendif}$
$_img1[PAD_IMG_ROWS][PAD_IMG_COLS];floatpad_img2[PAD_IMG_ROWS][PAD_IMG_COLS];floatpad_img3[PAD_IM$
$_for_rows:$
$for(inti=$
$0;i<$
$PAD_IMG_ROWS;++$
$i)clone_for_cols:$
$for(intj=$
$0;j<$
$PAD_IMG_COLS;++$
$j)pad_img1[i][j]=pad_img0[i][j];pad_img2[i][j]=pad_img0[i][j];pad_img3[i][j]=pad_img0[i][j];$
$_section(pad_img0,pad_img1,pad_img2,pad_img3,prediction);$
CNN
$_Result_1.pngfigure()$
$_Result_2.pngfigure()$
$_Result_3.pngfigure()$

${}_{p}redict_{1}00.pngfigureAccuracy$
${}_{p}redict_{5}00.pngfigureAccuracy$