Task1: Creating a small image processing library

By: **Rachit Kumar**-2017CS10364 cs1170364@iitd.ac.in
Rahul Choudhary-2017CS10365 cs1170365@iitd.ac.in

1 In this, we have implemented five main functions, namely:

- Convolution(with and without input padding, as convolution and as matrix multiplication(multiplication implemented using mkl, openblas, pthreads and iterative formula))
- Non-linear activations(relu and tanh)
- Subsampling(maxpooling and average pooling)
- Converting a vector of random floats to a vector of probabilities(softmax and sigmoid)
- Implementation of LeNet architecture

2 Input formats:

- **Convolution without input padding:**

  ./output convolution_withoutpadding matrix1.txt matrix1_numrows matrix2.txt matrix2_numrows

- **Convolution with input padding:**

  ./output convolution_withpadding padsize matrix1.txt matrix1_numrows matrix2.txt matrix2_numrows

- **Convolution without input padding as matrix multiplication:**

  ./output convolution_withoutpadding_matrixmult matrix1.txt matrix1_numrows matrix2.txt matrix2_numrows (method of matrixmult(manual/mkl/openblas/pthread))

- **Convolution with input padding as matrix multiplication:**

  ./output convolution_withpadding_matrixmult padsize matrix1.txt matrix1_numrows matrix2.txt matrix2_numrows (methodof multiplication (manual/mkl/openblas/pthread))

- **relu function on matrix elements:**

  ./output activation relu matrix.txt number_of_rows_in_matrix number_of_columns_in_matrix

- **tanh function on matrix elements:**

  ./output activation tanh matrix.txt number_of_rows_in_matrix number_of_columns_in_matrix

  In this function, we apply tanh function(i.e. f(x)=(exp(x)-exp(-x))/(exp(x)+exp(-x))) on each individual element of the given matrix.

- **Maxpooling of square matrices:**

  ./output subsampling maxpooling matrix.txt number_of_rows_in_input_matrix number_of_rows_in_output_matrix

- **Average pooling of square matrices:**

  ./output subsampling avgpooling matrix.txt number_of_rows_in_input_matrix number_of_rows_in_output_matrix

- **Sigmoid:**

  ./output probability sigmoid vector.txt size_of_vector

- **Softmax:**

  ./output probability softmax vector.txt size_of_vector

- **LeNet:**

  ./output lenet input.txt

3 Performance comparision:

- For very small inputs and kernel, the order of time taken is: OpenBLAS > Iterative Multiplication > pthread > MKL. As we increase the matrix size, MKL starts to become faster and faster and for large inputs, they take almost the same time.

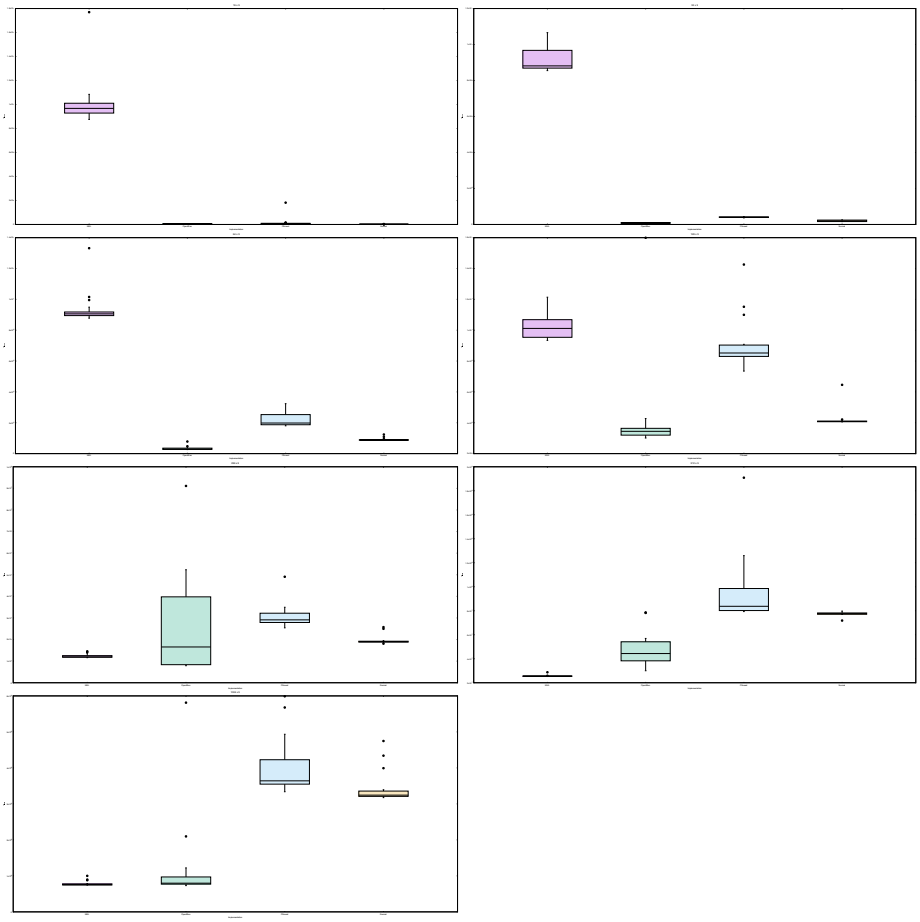
- pthread is initially much slower than normal multiplication but then for large inputs, it takes time comparable to that of normal multiplication. But on much larger inputs, it again starts to take much more time than the normal multiplication. Also, the performance of pthreads depends greatly on the number of cores in the machine on which the code is used. For best performance, the number of threads in the code should be equal to the number of cores in the machine(which in our case was two).

MKL	
size	**time(ns)**
16X16	$9.5x10^6$
32X32	$8.5x10^6$
64X64	$9x10^6$
128X128	$1x10^7$
256X256	$1.25x10^7$
512X512	$2.5x10^7$
1024X1024	$7.5x10^7$

OpenBLAS	
size	**time(ns)**
16X16	very low
32X32	10^5
64X64	$2.5x10^5$
128X128	$3.5x10^5$
256X256	$1.5X10^7$
512X512	$4.75x10^7$
1024X1024	$8x10^7$

pthread	
size	**time(ns)**
16X16	very low
32X32	$4x10^5$
64X64	$2x10^6$
128X128	$8x10^6$
256X256	$3x10^7$
512X512	$8x10^7$
1024X1024	$9.5x10^7$

Manual	
size	**time(ns)**
16X16	very low
32X32	$2.5x10^5$
64X64	$1x10^6$
128X128	$4x10^6$
256X256	$2x10^7$
512X512	$8x10^7$
1024X1024	$3x10^8$



4 Error Handling:

- In case of incorrect function argument, it displays the valid function arguments which can be given.
- In case of incorrect number of arguments, it displays the whole input format for the respective function.
- Compulsion on the number of rows and columns(or the length of vector) to be always positive.
- In case of convolution, error message is printed if the kernel size is larger than the size of the input matrix.
- In case of invalid file input or invalid data type input for size of arrays, an exception is thrown.

[1].
[2].

References

[1] *Intel MKL.*

[2] *OpenBLAS.*