**An Image Processing Library**

**WEEK 1:**

**Contents:**

1. Week1.cpp
2. MakeFile\_1.txt
3. Readme\_1.txt

**Steps:**

**Preferred Step:**

**Use Visual Studio C++**

**Alternative**

1. gcc Week1.cpp -lstdc++ -o Week1.o
2. ./Week1.o …. (Continuing the terminal statements for parameters)

**Commands:**

1. ./Week1 convolution padsize matrix1.txt num\_rows1 matrix2.txt num\_rows2

**Purpose - Convolution of matrix1 with filter matrix2**

If number of rows of matrix1 or matrix2 is not an integer or less than 0,an error is shown.if padsize is negative or not an integer, an error is show

1. ./Week1 convolutionMM padsize matrix1.txt num\_rows1 matrix2.txt num\_rows2

**Purpose - Convolution of matrix1 with filter matrix2 using matrix multiplication**

1. ./Week1 ReLU matrix.txt num\_rows

./Week1 tanh matrix.txt num\_rows

If number of rows is negative or not an integer, an error is thrown. Tanh function is used from the existing cmath lib in c++.

**Purpose - Applies ReLU or tanh operation on the given input matrix**

1. ./Week1 sigmoid vector.txt sizeof\_vector

./Week1 softMax vector.txt sizeof\_vector

If size of vector is negative or not an integer, an error is thrown.From cmath lib, the known mathematical value e is known.

**Purpose - Gives the sigmoid function/softMax of the input vector**

**Functions Used:**

**For convolution -** vector<vector<float> > convolution(int pad, vector<vector<float> > UnpaddedInput, vector<vector<float> > kernel)

**For convolutionMM –**

**1.** vector<vector<float> > matrixMultiply(vector<vector<float> > A, vector<vector<float> > B)

**2.** vector<vector<float> > scopedMatrix(vector<vector<float> > input,int scopedMatrixSize, int Row, int Column)

**3.** vector<vector<float> > squeezeMatrixToColumn(vector<vector<float> > input)

**4.** vector<vector<float> > flattenMatrixToRow(vector<vector<float> > input)

**5.** vector<vector<float> > Toeplitzise(vector<vector<float> > input, int KernelSize)

**For ReLU –**

1. vector<vector<float> > MatrixReLU(vector<vector<float> > input)
2. float ReLU(float num)

**For tanh –**

1. vector<vector<float> > MatrixTanh(vector<vector<float> > input)

**For Sigmoid –**

1. vector<float> sigmoid(vector<float> input)

**For SoftMax –**

1. vector<float> softMax(vector<float> input)

**For Average Pooling –**

1. vector<vector<float> > AveragePooling(int pad,int PoolSize, vector<vector<float> > UnpaddedInput)

**For Max Pooling –**

1. vector<vector<float> > MaxPooling(int pad,int PoolSize, vector<vector<float> > UnpaddedInput)

**For Matrix view –**

1. void printMatrix(vector<vector<float>> Matrix)

**For Vector view –**

1. void printVector(vector<float> Vector)

#Examples

./Week1.out softMax InputVector.txt 5

./Week1.out sigmoid InputVector.txt 5

./Week1.out tanh Kernel.txt 3

./Week1.out ReLU Kernel.txt 3

./Week1.out convolutionMM 0 InputMatrix.txt 5 Kernel.txt 3

./Week1.out convolution 0 InputMatrix.txt 5 Kernel.txt 3