A TensorFlow example

Kamilla Faizullina

We implement Conway's Game of Life using TensorFlow and run the program either using GPU or CPU.

1 Implementation

For the main function we use built-in conv2d function which applies a 2D convolution over an state of the simulation. To count information from neighbors cells, we can use kernel consisting of ones.

```
 kernel = tf.reshape(tf.ones([3,3], dtype=board.dtype), [3,3,1,1])
for i in range(iters):
 neighbours = tf.nn.conv2d(board, kernel, [1,1,1,1], "SAME") - board
 survive = tf.logical_and(tf.equal(board, 1), tf.equal(neighbours, 2))
 born = tf.equal(neighbours, 3)
 board = tf.cast(tf.logical_or(survive, born), board.dtype)
```

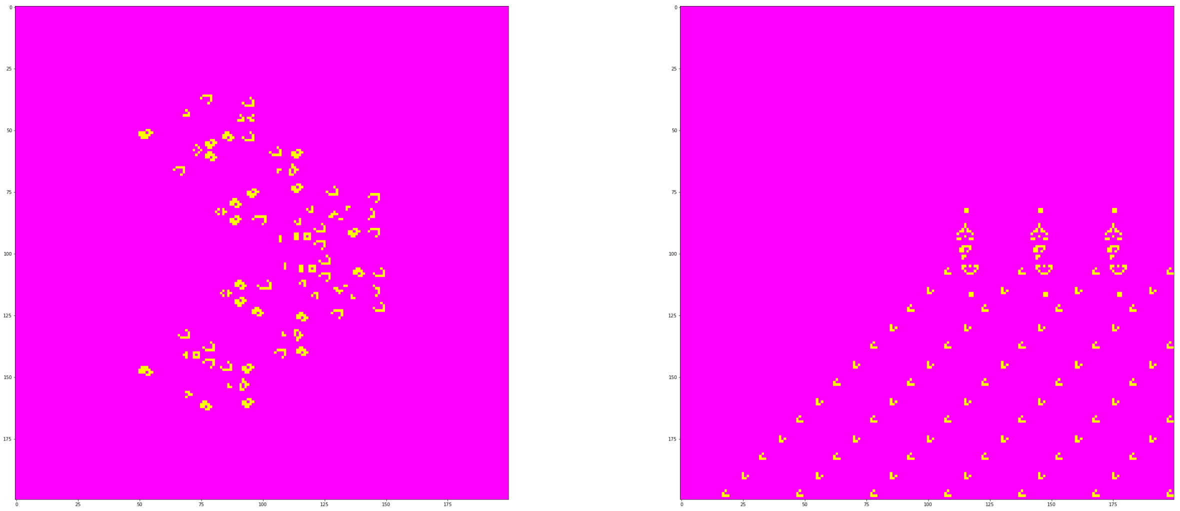
2 Performance for CPU and the GPU

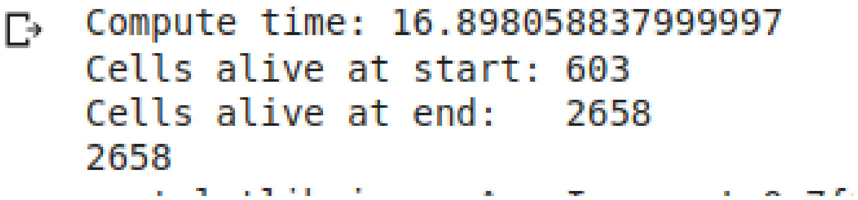
Tensorflow allows to run program on the GPU (I suppose it is quite optimized in comparison to my own implementations).

Type	GPU time	CPU time
tf.float16	12.461	49.87
tf.float32	11.22	40.872
tf.float64	12.685	44.837
tf.int32	19.8	50.715

Table 1: Results

Table 1 presents obtained run time for four different data types. In all cases, GPU had more performance.





```
Compute time: 16.898058837999997
Cells alive at start: 603
Cells alive at end:   2658
2658
```

Figure 1: First and Last states. I made tests on Google Collab to validate the program.