

- CS Login: **dspatel6**
- Wisc ID: **9085310937**
- Email: **dspatel6@wisc.edu** / dspatel6@cs.wisc.edu

CPU Specifications

```
- Processor 13th Gen Intel(R) Core(TM) i5-1350P    1.90 GHz
- Installed RAM 16.0 GB (15.4 GB usable)
- Total Cores 12
- Performance-cores 4
- Efficient-cores 8
- Total Threads 16
- Max Turbo Frequency 4.70 GHz
- Performance-core Max Turbo Frequency 4.70 GHz
- Efficient-core Max Turbo Frequency 3.50 GHz
```

Memory bandwidth

```
DDR5-4800 (Dual-Channel)
Memory speed: 4800 MT/s
Bus width per DIMM: 64 bits (8 bytes)
Channels: 2 (dual-channel)
4800 × 8 × 2 = 76.8 GB/s
```

Compiler

```
```bash
$ g++ --version
g++ (Ubuntu 14.2.0-16ubuntu1) 14.2.0
Copyright (C) 2024 Free Software Foundation, Inc.
This is free software; see the source for copying conditions. There is NO
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR
PURPOSE.
```
```

OS

```
```bash
$ cat /etc/os-release
PRETTY_NAME="Ubuntu Plucky Puffin (development branch)"
NAME="Ubuntu"
VERSION_ID="25.04"
```
```

Compilation command

```
```bash
$ make
```
```

Results for our implementation:

| Non-MKL | Resolution | 512 * 512 | 512 * 512 | 512 * 512 | 1024 * 1024 | 1024 * 1024 | 1024 * 1024 | 2048 * 2048 | 2048 * 2048 | 2048 * 2048 | 4096 * 4096 | 4096 * 4096 | 4096 * 4096 |
|---------|------------|-----------|-----------|-----------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
| | Block-size | 16 * 16 | 32 * 32 | 64 * 64 | 16 * 16 | 32 * 32 | 64 * 64 | 16 * 16 | 32 * 32 | 64 * 64 | 16 * 16 | 32 * 32 | 64 * 64 |
| Threads | 1 | 10.55 | 9.21 | 8.94 | 88.2 | 84.96 | 66.77 | 1208.73 | 774.56 | 607.33 | 11907 | 6712 | 4855 |
| | 6 | 4.21 | 4.41 | 3.51 | 27.97 | 20.43 | 20.62 | 256.268 | 191.44 | 157.79 | 2271 | 1382 | 1138 |
| | 12 | 2.33 | 2.54 | 2.43 | 18.2 | 15.35 | 15.489 | 189.11 | 129.01 | 115.765 | 1601 | 1020 | 924 |

Observations:

- In case of 512 * 512 resolution, changing block-size to higher block-sizes does not impact the performance significantly
- We see a huge performance increase when going from number-of-threads 1 to 6, but same is not true when moving from 6 to 12. This might be because we must be hitting memory bounds before we hit the compute bounds. It is easily evident from the 4086 * 4086 resolution matrix
- With higher resolutions of matrices, the block-size parameter has a higher impact on performance than the smaller resolutions
- In higher resolutions, there is a lot of performance which is left on the table which is achieved by using parallel compute

Results for MKL-based implementation:

| MKL | Resolution | 512 * 512 | 1024 * 1024 | 2048 * 2048 | 4096 * 4096 |
|---------|------------|-----------|-------------|-------------|-------------|
| Threads | 1 | 2.58 | 17.57 | 130.87 | 1015 |
| | 6 | 1.74 | 9.43 | 48.93 | 333 |
| | 12 | 1.23 | 7.37 | 54.5 | 333 |

Observations:

- In MKL-based implementation, we no longer see performance improvement when moving from 6 to 12 threads.