# 

Final project





and Internet Master's program in Informatics Engineering Things (IoT)

26610 Student: Darya Martsinouskaya

Computing Course Unit: High Performance

Jasnau Caeiro Teacher: José

## Introduction

K-means is a widely used clustering algorithm in machine learning.

**usters** minimizing intra-cluster distances It partitions datasets into k cl

significantly for large datasets [1]. computational cost grows The

strategies using OpenMP and GPU acceleration to improve efficiency tion The work explores paralleliza

### **Parallelization** Strategies

### Parallelization of main loops for point assignment and centroid OpenMP (Multicore CPU): updates

The standard algorithm iteratively

Sequential K-means:

description

**Problem** 

assigns points to clusters and

updates centroids.

## **GPU Acceleration:**

Offloading computations to GPU for enhanced parallel execution.

alability for large datasets (e.g.,

MNIST).

High time complexity limits

Challenges:

Reduce execution time while

God:

maintaining accuracy.

### Speedup Calculation: Speedup (S) = Time (Sequential) Time (Parallel)

### mental Setup Experi

(grayscale images, 28x28 pixels) NIST Dataset: M

Metrics: Execution time, accuracy, speedup

- Implementations: Sequential (baseline)
  - OpenMP parallelized GPU-accelerated 0
    - -accelerated

### Setup Experimental

05

# Dataset: MNIST (grayscale

images, 28x28 pixels)

Metrics: Execution time, accuracy, speedup

# Implementations:

- Sequential (baseline)
  - OpenMP parallelized
  - GPU-accelerated 0 0

### Results 90

Speedup GPU	663.55	1206.21	734.61	556.57
Speedup OpenMP	8.31	30.73	45.92	17.20
GPU Time (s)	1.26322	1.23969	1.2539	1.24277
OpenMP Time (s)	100.817	48.6627	20.0596	40.2045
Sequential Time (s)	838.213	1495.33	921.122	691.691
*	10	20	100	200

- GPU acceleration significantly outperforms both sequential and OpenMP dnpaads implementations, achieving up to 1206.21×
- OpenMP provides moderate improvements (up to 45.92× speedup) but loses
- efficiency at higher k values All implementations maintain identical clustering accuracy, confirming correctness

07

# Conclusion

- and GPU-accelerated versions of K-means Implemented sequential, OpenMP parallel,
- 45.92× speedup, but gains decreased for large OpenMP parallelization achieved up to cluster sizes
- improvement, reaching 1206.21× speedup with GPU acceleration provided the highest excellent scalability
  - implementations, validating parallelization Accuracy remained consistent across correctness
- management) could enhance performance for larger datasets (e.g., CUDA, memory Further optimizations

### References

AnjaTanovićandVukVranjković."ImplementationofparallelK-meansalgorithmfor image classification using OpenMP and MPI libraries". In: 2024 Zooming Innovation in Consumer Technologies Conference (ZINC). 2024, pp. 54–59. doi: 10.1109/ZINC61849.2024.10579351.