ICDCS Paper: *Energy-Aware CPU Frequency Scaling for Mobile Video Streaming* [DOI: 10.1109/ICDCS.2017.74]

- **Key findings –**
  1. The wireless interface states for LTE networks always stays in high consumption state.
  2. Video streaming uses a lot of CPU power to deliver a good quality of service. Energy consumption is a factor of CPU frequency.
  3. Data transmission energy and CPU energy both play an important role in deciding the phones total energy consumption while streaming.
  4. With decreasing LTE rates, it is becoming the main network for consumption of media. Which ultimately increases the battery consumption.

- **Key technology insights –**
  1. EFS algorithm combined with efficient download method can reduce energy consumptions up to 50% on the official YouTube app.
  2. Large energy consumptions with video streaming on mobile devices.
  3. Optimizing the downloading schedules is required to decide how much and when to download the video that needs to be played
  4. Different video resolutions and bit rate plays and important role in deciding the desired download data.

- **Relevance to CPU, GPU and Processor Scalability –**
  1. CPU frequency are significant while considering energy savings of 1000s of devices.
  2. With increase in more production efficient GPUs, the CPU power consumption for video processing can be reduced significantly.
  3. Mobile devices can be used as a P2P network to serve already available videos to nearby devices, improving the power efficiency of the network.
  4. Transmission power losses can be minimized while scaling network using efficient download methods like MaxMin.

Review Paper: *GPU Based Strategies for Distance-Based Outlier Detection* [DOI: 10.1109/TPDS.2016.2528984]

- **Key findings –**
  1. Outlier detection is one of the data mining functions for finding interesting patterns in large data sets.
  2. Outlier detection is very performance heavy and requires high computing power.
  3. GPUs can be used to perform parallel algorithms like SolvingSet and BruteForce in distributed environment for outlier detection.
  4. Complex outlier detection algorithms cannot be used for online applications which require minimum response time.

- **Key technology insights –**
  1. BruteForce algorithms can detect top n outliers, but are inefficient for online applications.
  2. NVIDIA CUDA fermi architecture is used for achieving parallel thread execution.
  3. K nearest neighbor queries by Kato and Hosino exploited for obtaining ton n outliers.
  4. Bay's parallel algorithm is used for outlier detection.

- **Relevance to CPU, GPU and Processor Scalability –**
  1. Distributed architecture is fully utilized with GPU-DistributedSolvingSet algorithm designed in paper.
  2. GPU algorithms for outliers can utilize parallel and distributed processing reducing the space cost and requirement.
  3. Recent development and wide availability of GPUs with hundreds of cores and parallel processing has facilitated new algorithms to exploit this architecture.
  4. Data partitioning with multi node processing significantly increases algorithm performance