Abstract:

The coming decade will be the era of smart, intelligent systems; many of these will have computer vision algorithms at their core. Such systems include self-driving cars, UAVs, robots, mobile nodes with edge computing support, and AR/VR systems. Hence, accelerating computer vision workloads is a very important problem as of today and has begun to attract a considerable amount of attention in both academia and industry. As of 2021, the community appears to have significantly migrated towards CNN (convolutional neural network) based algorithms that run on GPUs and custom accelerators including FPGAs. A significant proportion of state-of-the-art systems comprises highly optimized CNN-based algorithms running on bespoke accelerators and GPUs. This trend is justified because such design choices have proven their
mettle against traditional algorithms that primarily relied on multicore processors for parallelization.

However, we argue that this trend is showing signs of saturation, does not hold all the time, and may not prevail in the future. We show that as the number of concurrent applications (conventional vision algorithms) running on GPUs increases, the performance drops due to increased contention of the shared resources. In contrast, the performance of a bag of these concurrent applications on multicore processors scales well with an increase in the number of concurrent applications due to well-developed contention management policies in multicores and the inherent nature of conventional vision algorithms. This work establishes multicore CPUs as legitimate systems for executing conventional vision applications. Based on our characterization, we also identify a unique phase behavior in the execution of these algorithms and leverage it to propose an auction theory-based scheduling algorithm that gives us near-optimal schedules very quickly without sacrificing fairness.

We then characterize the CNN-based algorithms and find that the excessive tilt towards these algorithms is not justified. We show that for inclement weather scenarios, traditional stereo algorithms are sometimes more accurate than CNN-based stereo algorithms. Hence, an ensemble of these algorithms should be deployed to improve the robustness of these systems, especially on safety-critical systems such as self-driving cars. Lastly, we realize that these computer vision systems are mostly deployed as a part of bigger systems such as UAVs. Apart from the right choice of the computer vision algorithm, there are many more parameters that are important such as collision-free path planning, power efficiency, and flight duration. We show that when we include all these parameters with the vision system, optimizing the full set of parameters for the entire system becomes a very complex problem. It is large, multi-dimensional, non-convex at many places, and not amenable to solution in real-time. We use game theory again and replace this intractable optimization problem with a simpler yet approximate game theory equivalent: a set of players with different payoffs and strategies.