

Implementasi Penggunaan Kubernetes Cluster sebagai Infrastruktur Aplikasi Berbasis Web untuk Kuantifikasi Kepadatan Lalu Lintas Secara Real Time = Implementation using Kubernetes Clusters as a Web-based Application Infrastructure for Quantification of Traffic Density in Real Time

Muhammad Farhan Almasyhur, author

Deskripsi Lengkap: <https://lib.ui.ac.id/detail?id=20518461&lokasi=lokal>

Abstrak

Salah satu fungsi CCTV yaitu untuk menjaga lalu-lintas pada persimpangan jalan, yang merupakan bagian dari ATCS. Ketika semakin banyak ATCS yang terpasang di jalan, secara konvensional, semakin banyak layar yang harus dipantau pada control room. Hal ini memerlukan sumber daya manusia tambahan, misalnya untuk dapat mengontrol keseluruhan lampu merah di persimpangan jalan. Sehingga, penggunaan algoritma YOLO dapat membantu melakukan pendeteksian kepadatan lalu lintas pada suatu persimpangan secara otomatis. Namun, menjalankan proses ini di sebuah server dapat mengakibatkan performa yang buruk bila jumlah aliran video dari CCTV yang harus diproses bertambah. Penelitian ini melakukan proof of concept untuk implementasi sistem di atas secara lebih scalable. Video yang berasal dari beberapa CCTV dikirimkan ke sebuah cluster kubernetes dengan memanfaatkan arsitektur multi-core pada GPU. Proses ini menghasilkan video deteksi yang telah diberikan bounding box dari kendaraan yang berada dalam lalulintas. Selain itu, terdapat hasil dari hitungan banyak kendaraan dari masing-masing lajur yang disimpan dalam sebuah databas. Data tersebut digunakan untuk web dashboard yang digunakan untuk memudahkan petugas dalam melakukan pemantauan dan pengambilan keputusan. Uji coba yang dilakukan membuktikan bahwa sistem dapat bekerja dengan stabil dengan auto-scaling mengikuti jumlah workload, dengan rerata penggunaan CPU yaitu 490 mCore, rerata penggunaan memory sebesar 1,7 GB untuk masing-masing pod, rata-rata penggunaan GPU 1,2 GB untuk satu koneksi client, dan 2 GB untuk dua koneksi client.

.....One of the functions of CCTV is to maintain traffic at crossroads, which is part of ATCS. As more and more ATCS are installed on the road, conventionally, more screens will have to be monitored in the control room. This requires additional human resources, for example, to be able to control the total number of red lights at crossroads. Thus, the use of the YOLO algorithm can help detect traffic density at an intersection automatically. However, running this process on a server can result in poor performance if the number of video streams from CCTV has to increase. This research does a proof of concept to implement the system in a more scalable way. Video from multiple CCTVs is sent to a Kubernetes cluster by leveraging the GPU's multi-core architecture. This process produces a detection video that has been assigned a bounding box from that in traffic. In addition, there are results from the count of the number of vehicles from each lane that are stored in a database. The data is used for a web dashboard that is used to make it easier for officers to monitor and make decisions. The tests were carried out to prove that the system can work stably with automatic scaling according to the number of workloads, with an average CPU usage of 490 mCore, the average memory usage of 1.7 GB for each pod, an average GPU usage of 1.2 GB for one client connection, and 2 GB for two client connections.