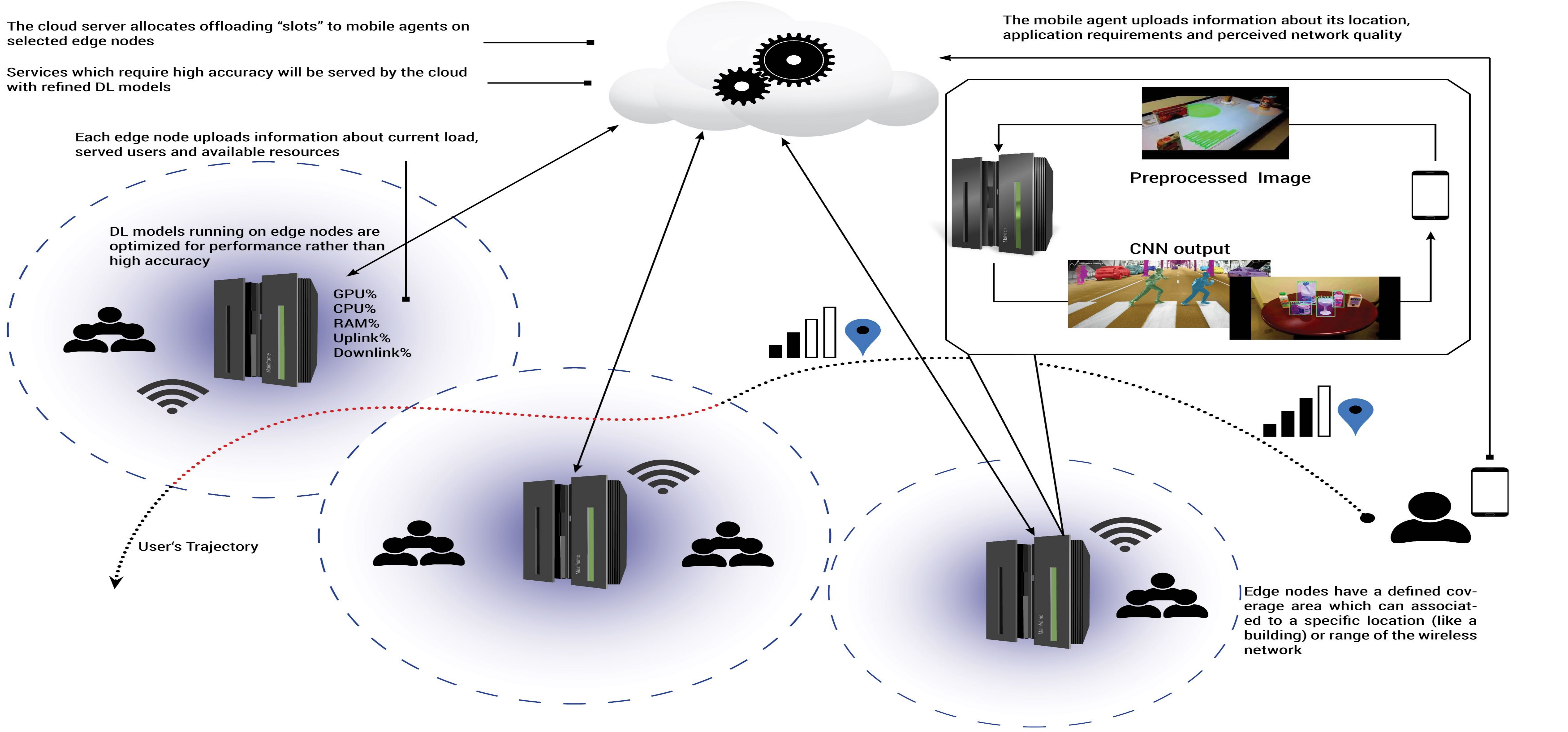
mDOCS: Mobile Deep-Neural-Networks Offloading Edge-Cloud Scheduler

Vittorio Cozzolino, Leonardo Tonetto
{vittorio.cozzolino; tonetto}@in.tum.de
Chair for Connected Mobility



Overview

- **Computer Vision** powers every mobile Augmented Reality (AR) application available on mobile devices
- Machine Learning for mobile AR applications increases **immersiveness** (e.g. Pokemon GO, PinAR):
  - SLAM (sparse or dense), SIFT
  - Image classification
  - Object recognition
  - Image segmentation
  - Panoptic segmentation
- Such models either run directly on the phone (downgraded version) or make use of cloud resources

Problem

- Smartphones are either not powerful enough or the applications too demanding in terms of energy drain
- Support old smartphone which do not have GPU acceleration, AI chips or octa-core CPUs
- Mobile Data Traffic is often expensive
- AR applications are highly demanding on data traffic
- Deep Learning models on mobile devices deliver reduced accuracy due to simplifying optimization steps like pruning, layers fusion, quantization etc.

Goals

- High performance mobile AR experience in a multi-tenant scenario regardless of mobile devices hardware
- Provide a scheduling algorithm based on a multivariate, multi-constrained heuristic with a trifecta target function:
  - maximization of throughput (FPS)
  - minimization of latency
  - extension of battery life
- Proactively allocate resources as users move in-and-out from edge nodes' coverage area
- Optimize inter-node scheduling for efficient GPU sharing

Our Approach

- Exploit nearby infrastructure by offloading computation to hardware:
  - In the **Cloud**
  - At the **Edge**
- Optimize for network condition, mobility, available hardware resource, required accuracy, etc.
- Centralized decision making for better resource allocation

Edge-Cloud Scheduling

- Centralized, cloud-based (overseer) which decides:
  - How/which edge node should serve a mobile agent in proximity following our heuristic
  - when to offload at the edge, when to use the cloud or if to just run everything locally
- Edge nodes and mobile agents periodically send updates to notify their current status (e.g. position, current load)
- Two macro-sets of parameters influence:
  - Network and Hardware constraints
  - Human Mobility constraints

Network and Hardware Constraints

- Available edge nodes resources: GPU, CPU, RAM
- Network conditions and fluctuations
- Application requirements: accuracy, recall, speed
- Impact of multi-tenancy
- Performance impact of sharing a GPU across multiple users
- Heterogeneous devices: Jetson Nano, TX2, server GPUs
- High-precision networks vs quantized optimized networks

Human Mobility Constraints

- User's position and network coverage (WiFi/Mobile Network)
- Trajectory prediction for preemptive resource allocations
- Mobility and Data Traffic must show strong correlation*