Offre de stage 2021-2022

Titre	**Decentralized live video analytics at the network edge**
Niveau du stage	Master 2ème année / Ingénieur 5ème année
Date de début/ fin	De février-mars 2022 au juillet 2022 (4-6 mois)
Ville, Pays	Annecy-le-Vieux, *France*
Laboratoire	LISTIC - Laboratoire d'Informatique, Systèmes, Traitement de l'Information et de la Connaissance - http://www.polytech.univ-savoie.fr/LISTIC
Description du sujet de stage	**Context.** Extracting information from cameras is increasingly becoming vital across a number of applications. For example, cities have been deploying a large number of closed-circuit cameras that are used for safety, security, and traffic control applications. For example, the police department of New York City uses over 18 thousand cameras spread all over the city [1]. Live video analytics solutions [2] rely on distributed cloud resources to implement pipelines of operations that process incoming video frames captured by these cameras to progressively extract relevant information from them. Until now, advances in video analytics have largely focused on the use of fixed cameras, particularly in the domain of traffic surveillance, that observe the same scene during their lifetime (e.g., [3, 4, 5]). Analytics architectures process video streams from these cameras through a hierarchy of processing clusters and a somewhat predictable path in the network. They are part of an organized and well-understood deployment that provides a lot of control knobs at each tier within the defined hierarchy. Yet, with the easier-than-ever access to and rising penetration of mobile cameras, standard video analytics architectures should adapt to incorporate their video feeds. These cameras, mounted on devices such as smartphones, car dashboard, drones, etc., often offer the unique advantage of being in the right place at the right time. However, in the absence of real-time support for gathering analytics from them, we are missing an opportunity for truly pervasive intelligent analytics. The DIVA project aims to develop the architectural support required to introduce mobile video cameras into the video analytics platform. **State of the art.** In recent years, video analytics have become a key to supporting surveillance, security, traffic control, and multimedia use cases with processing at the edge of the network [2]. Different solutions have been proposed to support dynamic extraction of useful information from live camera feeds. Spatula [3] uses temporal and spatial correlations between video streams from large camera networks to create a cross-camera analytics platform. It uses a pre-learnt model of cross-camera correlations to reduce the time taken for spatio-temporal search among the video streams. Similarly, Chameleon [4] adapts its video pipeline parameters like the employed models to the current scene of a video stream to find the best balance between accuracy and speed. Hetero-Edge [5] first proposed an architecture that distributes computing tasks across nodes exploiting the inherent parallelism of tasks belonging to the analytics pipelines. The platform divides input tasks among slave nodes using the task's Directed Acyclic Graph and employs a scheduler that can make parts of frames independently processed by different slave nodes concurrently in favorable conditions. Finally, VideoEdge [6] proposes an optimization framework to determine the video resolution of the stream and placement of functions in different clusters. However, VideoEdge proposes a hierarchical distribution only, which does not adapt to the dynamic needs of mobile cameras. In this project we aim to build on the insights presented in previous solutions to further push video analytics architectures towards exploiting all available resources to better address the challenges introduced by the integration of mobile cameras into the video analytics pipeline. We investigate the impact that horizontal distribution can have on resource consumption. **Project goals.** In this project, we argue that, to move to the next step in live video analytics and support mobile video sources, compute architectures should maximize how they distribute their processing pipelines. Taking advantage of ultra-localized computing nodes, orchestrators gain the ability to split processing pipelines not only hierarchically but horizontally, too. Edge computing frameworks are rapidly evolving to include processing accelerators—i.e., GPUs [7] and TPUs [8]—at the extreme edge of the network. Such solutions aim to offer an extremely scalable approach to support low latency computation services

	through a pervasive deployment approach. Unfortunately, these architectures are inherently underpowered and relying on them to support video analytics requires to carefully decide how to distribute the processing load. This becomes particularly challenging when processing videos generated by mobile cameras as incoming traffic loads can rapidly shift over time. To support our hypothesis and answer the aforementioned question, we developed a first prototype for experimenting with distributed video analytics on edge computing nodes. Our initial results [9], suggest that smartly distributing the functions is an efficient way to utilize the resources at the edge. **Internship tasks.** Starting from the initial work previously performed, the hired student will work on the development of the orchestration architecture. Such architecture will focus on the ability to deploy vision functions across heterogeneous edge devices and rapidly adapt to ever changing introduced by cameras mobility. While the current implementation relies on manual configuration and deployment, we will explore how existing architectures (e.g., Kubernetes) could be extended to reliably work at the network edge. Further, the student will work on developing initial solutions for selecting how to deploy tasks on the available resources exploiting results from an empirical characterization of video analytics applications and how they are impacted by a number of quantifiable factors (e.g., resolution, frame rate, etc.) as well as by less deterministic ones (e.g., how many objects are present in a given frame). **References.** [1] Rocco Parascandola. 2018. New NYPD surveillance cameras to cover stretch of Upper East Side not easily reached by patrol cars. https://bit.ly/3kiiUCN [2] Ganesh Ananthanarayanan et al. Real-time video analytics: The killer app for edge computing. Computer 50, 10 (2017), 58–67. [2] Ganesh Ananthanarayanan et al. Project Rocket platform. Microsoft Research Blog. https: //www.microsoft.com/en-us/research/publication/project-rocket-platform-designed-for-easy-customizable-live-video-analytics-is-open-source/ [3] Samvit Jain et al. Spatula: Efficient cross-camera video analytics on large camera networks. In ACM/IEEE Symposium on Edge Computing (SEC 2020) [4] Junchen Jiang et al. Chameleon: Scalable Adaptation of Video Analytics. In Proceedings of the 2018 Conference of the ACM Special Interest Group on Data Communication (SIGCOMM ’18) [5] Wuyang Zhang et al. Hetero-Edge: Orchestration of Real-time Vision Applications on Heterogeneous Edge Clouds. In IEEE INFOCOM 2019. [6] Chien-Chun Hung et al. VideoEdge: Processing Camera Streams using Hierarchical Clusters. In 2018 IEEE/ACM Symposium on Edge Computing (SEC). [7] jetson 2021. Jetson Nano Developer Kit. https://bit.ly/3nDLxw5 [8] Sam Sterckval. Performance comparison: Coral Edge TPU vs Jetson Nano. https://bit.ly/3tJfegq [9] Sri Pramodh Rachuri et al. Decentralized Modular Architecture for Live Video Analytics at the Edge. To appear in the 3rd Workshop on Hot Topics in Video Analytics and Intelligent Edges (HotEdgeVideo 2021). [10] Francescomaria Faticanti et al. The Case for Admission Control of Mobile Cameras Into the Live Video Analytics Pipeline. To appear in the 3rd Workshop on Hot Topics in Video Analytics and Intelligent Edges (HotEdgeVideo 2021).
Compétences requises	• Comfortable speaking English or French (French is not required). • Good understanding of at least one between computer network protocols and systems or machine learning / vision methods (preferably both) • Good proficiency with at least one programming language, preferably Python, Golang, or Rust
Gratification	Selon législation en vigueur
Tuteurs / Contacts	Francesco Bronzino – fbronzino@univ-smb.fr

LISTIC