Embedded vision is probably at the edge of phenomenal expansion. The smart cameras are embedding some processing units which are more and more powerful. Last decade, high-speed image processing can be implemented on specifically designed architectures [1] nevertheless the designing time of such systems was quite high and time to market therefore as well. Since, powerful chips (i.e System On Chip) and quick prototyping methodologies are contently emerging [2],[3],[4] and enable more complex algorithms to be implemented faster. Moreover, smart cameras which are embedding flexible and powerful multi-core processors or Graphic Processors Unit (GPU) are now available and can be considered as well as a solution to implement faster some complex image processing algorithms. The smart camera can be considered as a powerful sensor which enables very complex information to be extracted in real-time from the video scene. Using several cameras simultaneously and dealing with a multi-view configuration is even more challenging but enable more information to be available. Therefore, we present in this paper a platform, named WiseEye, to manage and experiment on a smart camera network based on low-cost multi-core processors. A network of low-cost multi-core processors has been deployed. We have already developed a framework to ease application development and debugging [5]. The framework aims to offer a high degree of flexibility for managing the tasks and the memory allocation. Hence, the framework enables the priority and the allocation of each task to be controlled. The image acquisition task can be completely independent from the other processing tasks. The framework includes an interface dedicated to the management of the tasks: the user can add or suppress a task during the runtime, logs or processing results can be visualised for each task. Smart cameras use a dedicated network configuration and service providing tool named pym2SL (python Mesh and Multicast Services for Linux) which has been developed in le2i. pym2SL allows dynamic application management and configuration from a master node on the network, and service deployment and discovery at runtime. Services can be anything from video stream to processing resources, each allocable according to the user’s requirements. pym2SL is a free software, publicly available [6] under AGPL v3 open source license. The viewer application has been designed using multi-platform libraries only, it is based on a Qt GUI. The received video streams can be decoded with a software decoder or with hardware to reduce the CPU usage. The video streaming is based on GStreamer which offers high performances (TCP or UDP protocols available) with very low latencies.

We are currently investigating on the real-time implementation of complex image processing on this kind of hardware targets to provide extra services and security to the people living in an environment equipped with a smart camera network. Different targeted applications are then finally presented such as human tracking [7] for smart building management, control access based on a multi-modal approach, real-time fall detection [4] or recent remote physiological measurements (i.e. heart rate) based on video imaging [8].

**Keywords—Smart Camera, Multi-core processor, GPU, Real-time Image processing, photoplethysmography, fall detection, control access, smart building**

**ACKNOWLEDGMENT**

This research was supported by the Conseil Regional de Bourgogne Franche-Comte, France and the Fond Européen de Développement Regional (FEDER), and institut Carnot ARTS. The authors would like to thank Ali Douiyek and Duan Luguen for their contribution on this work.

**REFERENCES**


https://bullekeup.github.io/pym2sl
