Adaptive Energy-Oriented Multi-Task Allocation in Smart Camera Networks

Christos Kyrkou, *Member, IEEE,* Christos Laoudias, *Member, IEEE,* Theocharis Theocharides, Senior *Member, IEEE,* Christos G. Panayiotou, *Senior Member, IEEE,* Marios Polycarpou, *Fellow, IEEE*

Abstract—Computer vision applications of Smart Camera Networks (SCNs) often require that the network cameras operate under limited or unreliable power sources. Therefore in order to extend the SCN lifetime it is important to manage the energy consumption of the cameras which is related to the workload of the vision tasks they perform. Hence by assigning vision tasks to cameras in an energy-aware manner it is possible to extend the network lifetime. In this paper we address this problem by proposing a market-based solution where cameras bid for tasks using an adaptive utility function. The early results for different SCN configurations and scenarios indicate that the proposed methodology can increase network lifetime.

Index Terms— Embedded Vision Systems, Smart Camera Networks, Task Allocation, Adaptive Systems

I. INTRODUCTION

Smart camera networks (SCNs) have a wide application spectrum ranging from security and surveillance, transportation systems, healthcare and industrial monitoring [1],[2]. As the cost of cameras is rapidly reduced, and deployment of several cameras is within feasible costs, many smart cameras will be able to collaborate to carry out different vision tasks in dynamically changing environments [3]. These tasks need to be performed under time-varying constraints such as energy, computational resources, and performance. Hence, the network must be able to reconfigure over time to adapt to these changes [4].

In particular, energy is a key constraint that is of high importance in embedded vision systems especially in the case of battery-operated cameras or in environments with little infrastructure (e.g. limited power supply). Hence, it is necessary to efficiently utilize the available resources in a SCN to preserve energy and maximize the lifetime of the network. As such, the vision tasks that the network cameras need to perform must be allocated amongst them in an energy-oriented manner that simultaneously satisfies task-specific requirements such as resolution and frame-rate. Attempts to tackle the problem of energy-aware task allocation assume that all cameras execute the same task (usually target tracking) [10]. In cases where multiple tasks can be executed by the cameras energy is not considered as a key task assignment constraint [8].

The contribution of this paper with regards to the problem of energy-oriented multi-task allocation in heterogeneous SCNs is twofold. First, we formulate the problem and propose a distributed and adaptive market-based bidding solution that simultaneously considers multiple factors to determine which camera is most appropriate to satisfy a certain vision task. Second, a mechanism is proposed that enables a camera to change its bidding strategy according to its current state to further increase its lifetime. The proposed approach is evaluated using a fully simulated environment in MATLAB with different camera setups consisting of heterogeneous cameras and dynamic as well as static vision tasks. Preliminary results indicate that the lifetime of SCNs can be improved between $17\% - 64\%$.

The paper is organized as follows. Section II discusses related work. Section III presents the proposed distributed market-based approach. Section IV discusses the evaluation process and results, while Section V concludes the paper.

II. RELATED WORK

Dynamic reconfiguration and task allocation has been the subject of emerging research in SCNs. A centralized evolutionary algorithm has been proposed in [6] that simultaneously addresses area coverage and task assignment. It uses a two-step approach to first select cameras for area coverage and then assign tasks to cameras by considering only the frame-rate requirements of the tasks but not the energy resources of each camera. In [7], authors propose a decentralized approach for task assignment formulated as a stable marriage problem. The proposed approach is more suitable for one-to-one task assignment. Furthermore, energy is again not considered as a assignment criterion. A distributed algorithm that simultaneously considers area coverage of the cameras and subsequently target handover is presented in [8]. However, it only considers the tracking task and only for a small number of objects that can move within the area of observation of homogeneous cameras. The centralized and distributed approaches outlined in [9] attempt to reassign targets to cameras while managing their resources and also try to minimize the communication between them, rather than minimizing energy consumption to maximize their lifetime. Finally, the work in [10] attempts to perform energy-aware task assignment; however, it focuses only on target tracking with homogeneous cameras. Summarizing, most existing works assume that cameras are homogeneous, execute only the tracking task, and do not consider energy levels for task allocation purposes. In contrast, in our work we incorporate camera energy and resources with resolution and performance requirements in an adaptive distributed market-based task allocation framework in order to extend the lifetime of SCNs,

which are comprised of heterogeneous cameras that can execute different vision tasks with varying requirements.

III. PROBLEM FORMULATION

A smart camera network is considered where there are N_C static camera nodes C_j that belong in the set $\boldsymbol{C} = \{C_1, C_2, ..., C_{N_C}\}$ in a $2D$ field (Fig. 1), and N_V vision tasks T_i in the set $\boldsymbol{T} = \{T_1, T_2, ..., T_{N_V}\}$ that need to be performed and each is related to either a target or area. We are interested in the assignment $\boldsymbol{T} \rightarrow \boldsymbol{C}$, that will lead to the output set: $\{(C_j, T_i)\} \, for \, 1 \leq j \leq N_C \, and \, 1 \leq i \leq N_V$, which is the allocation of tasks to cameras. This assignment is dynamically updated every iteration t as the tasks that must be performed in the network and camera resource and energy levels change over time. In this work we are interested in the assignment that maximizes the mean network lifetime defined as $\sum_{j=1}^{N_c} L_j / N_C$, where L_j is the lifetime of each node C_j (i.e. the number of iteration steps where a node is alive). We propose a solution based on a market-based bidding process where the cameras bid for each task based on their current state and suitability.

A. Camera Model

Each camera C_j in the network is modeled with a state $S_j = (E_j, R_j)$ that describes its remaining battery energy levels, and available computational and memory resources respectively. In general, a smart camera is equipped with an embedded processor with multitasking capabilities [6]. Hence, the definition of resources can be different depending on the desired granularity. It can represent CPU cycle slots, threads, or computing cores and memory. As such, we do not explicitly use a specific type of resource but rather resources are abstracted. Therefore the underlying architecture of each camera node may be different; and we assume that each camera can estimate its available resources and translate them to a common reference point. Furthermore, given tasks resource requirements, a camera C_j can calculate the resulting frame-rate F_{ji} it can provide task T_i. It can also calculate the resolution D_{ji} at which it views a task-related area/target. The network can consist of heterogeneous cameras that have different features such as different Field-of-View (FoV), different energy levels, or different capabilities in terms of resources. Finally, the cameras that can communicate with each other, as well as camera image correspondences and target associations are considered known and available.

B. Task Model

Two different types of tasks are considered, namely static and dynamic. Static tasks (e.g. monitoring of restricted area) are associated with a specific area and are executed for a certain time after which they are considered obsolete. Also we assume that the tasks have no dependencies between them and that the vision tasks are suitable for distributed embedded processing and we in this work we primarily focus on the task assignment problem. Dynamic tasks, such as target detection and tracking, are associated with a moving target that can change its position resulting in being viewed by different cameras. Each vision task T_i is characterized by different required resources, which each camera can translate to its own
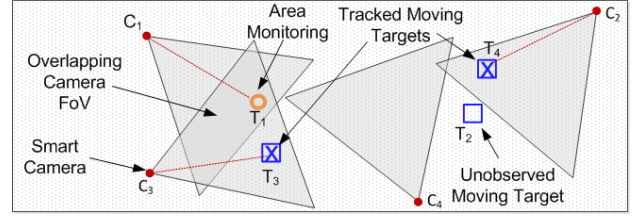

Fig. 1. Model setup example: C_1 is assigned static task T_1, C_3 is assigned dynamic task T_3, and C_2 is assigned dynamic task T_4. Task T_2 is unassigned, while C_4 is idle.

definition, and frame-rate. The resource requirements, complexities, and frame-rates of vision tasks are well documented and thus they can be configured a priori (e.g. in a look-up table). Finally, the location in x, y coordinates of each task-related area/target can be estimated by each camera using calibration and ground plane information [5], to compute the resolution D_{ji} at which a camera views a task related area/target [5].

C. Power Model

The power consumption model of each camera is based on the computing and memory resources used for running vision task(s) and for communicating with other cameras [11]. In addition, it is assumed that a fixed amount of power is consumed at every time instance t in order to perform necessary background tasks. Hence, we use the following energy consumption model for each camera C_j to find the energy consumed (E_j^c) by each camera at iteration step n.

$$E_j^C(t) = P_I + P(R_j) + P(m) \tag{1}$$

where $P(R_j)$ is a look-up based function that provides the energy consumption estimate based on utilized resources, $P(m)$ is also a look-up based function that gives the power consumption for transmitting m messages to other cameras, and P_I is constant power used when the camera is idle. Also, as in other works, we assume that the processing power is comparable to communication power [14]. The remaining battery level of a node is thus given by:

$$E_j = E_j^T - \sum_{t=0}^{n} E_j^C(t) \tag{2}$$

where E_j^T is the total energy capacity of camera C_j, and the second term is the sum of the energy consumed at each time iteration up to the current one. The subtraction gives as the remaining energy of each camera.

D. Distributed Market-Based Multi-Task Bidding Process

In distributed market-based approaches, task assignment happens by means of auctions where agents (cameras) play the role of bidders to gain an item (tasks) [3], [13]. This process is facilitated using the concept of *utility function*, which measures the ability and willingness of a camera to execute a certain task. Utility is a function that assigns a number (priority) to each camera such that if the utility is higher ($U_a > U_b$) then that camera is preferred. The utility function $U_j(T_i)$ (3) of camera C_j for task T_i is given by a weighted sum of four factors which encapsulate the state of the camera as well as its suitability to meet the requirements of a task. These are the remaining energy E_j of a camera, its available resources R_j; the frame-rate F_{ji} it can provide to the task and the resolution D_{ji} it views a task-related area/target.

Each component is normalized between $[0,1]$. This specific utility function was chosen due to its computational efficiency.

$$U_j(T_i) = W_E E_j + W_R R_j + W_F F_{ji} + W_D D_{ji} \qquad (3)$$

Prior to the actual task allocation the cameras first detect areas/targets of interest and exchange image descriptors in order to establish image correspondences and common FoVs. Then the cameras can coordinate to execute area/target related tasks such as intrusion detection and target tracking. The task allocation bidding process happens in an event driven manner where a bid for a task is initiated if there is an activity change in the FoV of a camera or if a camera is no longer operational. Once the task allocation process (*Algorithm 1*) is initiated each node that views a task-related area/target will have to evaluate its utility function $U_j(T_i)$ and also send a bid to the other nodes that view it. Each camera compares its bid with the received ones. The highest bidder takes over the task and updates its state and parameters while notifying the other cameras. A camera does not participate in the bidding process if it has no remaining available resources, no visibility of the task-related area/target, or no more remaining energy. In the current formulation, the bidding process happens iteratively for each item, however, without loss of generality combinations of items can also be considered.

E. Adaptive Strategy Selection

To enhance the performance of the market-based event-driven multi-task bidding process we also propose an adaptive energy-oriented method for cameras to change their bidding strategy (*Algorithm 2*). The approach is motivated by the fact that initially the nodes have full energy resources and thus can focus on satisfying the task demands, while only at the latter stages of their lifetime they can gradually give more importance to the energy factor in order to conserve energy. To do this we first introduce two energy thresholds used by all cameras (θ_1^E and θ_2^E). While the energy levels of a camera remain above the first threshold the initial values for the weights (w_E, w_R, w_F, w_D) are used and the goal is to give more significance to the performance components of the utility function. Once, the energy drops below the first threshold the weights are adjusted so that the energy and resource factors in the utility function are given equal or higher importance than the performance factors in (3). Finally, once the energy drops below the second threshold the impact of the last two terms in (3) is further reduced. The weights can have different initial values and the update method can vary. In this work, at each threshold step the energy and resource weight values are halved. Furthermore, in cases where the initial value of the energy and resource weights is zero (e.g. Set 3 Table II), then the weights of the other two factors are halved and simultaneously the energy and resource weights are increased. This dynamic adaptation allows cameras to have different task assignment strategies depending on their operating environment, battery levels, and workload. Hence, it offers a balanced way of changing the objective of each camera, depending on its current state, in order to extend the overall network lifetime.

Algorithm 1: Task Allocation

For $T_i \in \mathcal{T}$ and *event occurs*
  If camera C_j can execute *task T_i* **then**
    Multicast $U_j(T_i)$ to cameras $C_k \, \forall \, k \neq j$ that also see the *area/target*
    Receive $U_k(T_i)$ $\forall \, k \neq j$ of cameras that also see the *area/target*
    If $U_j(T_i) > U_k(T_i) \forall \, k \neq j$ **then**
      Camera C_j takes over Task T_i
      Update R_j, E_j
    end
  end

Algorithm 2: Strategy Selection

For $C_j \in \mathcal{C}$
  If $\theta_1^E < E_j \leq E_j^T$ *Maximum Performance State*
    weights = Initial_Weights;
  end
  If $\theta_2^E < E_j \leq \theta_1^E$ *Balance State*
    Halve Values of W_D W_F
    Increase Values of W_E W_R , if initial value was zero
  end
  If $E_j \leq \theta_2^E$ *Energy Conservation State*
    Halve Values of W_D W_F
    Increase Values of W_E W_R , if initial value was zero
  end

IV. EVALUATION & RESULTS

To evaluate our proposed energy-aware task allocation approach we have developed a visual simulation environment in MATLAB. Within a $2D$ field, N_C smart cameras with different battery levels and resources, and sensing range of 15 units (and increments of this), are generated either randomly, or under a specific pattern (Fig. 3) in an attempt to maximize camera overlap in a 200×200 unit area. We also assume that all cameras are capable of communicating given that their fields overlap. In turn, random targets are generated that follow predetermined paths (e,g, staircase, straight line) with constant speed. In addition, observation areas corresponding to static tasks that have a random start and end time and also have different requirements regarding resources and frame-rate are also generated within the field, resulting in a total of N_V vision tasks that need to be performed within the network. In total 11 different camera network setups (Fig. 2) were generated based on predefined structures and also random camera placement. The different network setups represent different scenarios where cameras can be used to monitor areas/rooms or transportation lines and crossroads, and random placements. The number of tasks also varied between scenarios, with an increasing number of tasks generated as the number of cameras increased as the proposed algorithms depend on the number of tasks and overlapping cameras in the network. We compare the proposed method for the same networks and setups with what we refer to as the baseline approach which uses the weights in Row 1 in Table II, and tasks are allocated in a periodic fashion and only based on the resolution and performance factors of (3), and without updating the weight values. The different parameter values are shown in Table I, while Table II shows the different initial weight values (Rows $2-4$) used by the proposed approach.

Various factors affect the performance of the energy-aware task allocation method such as the network structure, overlap between the FoV of the cameras, and the number and location of observation targets/areas, as well as the weight and energy threshold values. Networks with limited activity in the FoV of

TABLE I: Network Setup Information & Parameters

Number of Cameras (N_C)	6-30	Number of vision tasks (N_V)	30-90
Vision Tasks Resource Requirements (%)	5-25	Vision Tasks Performance Requirements (FPS)	6-30
Energy Unit Consumed Per Processing Resource	0.1	Energy Unit Consumed Per Messaged Transmitted	0.08

TABLE II: Different Initial Weight values in Different Experiments

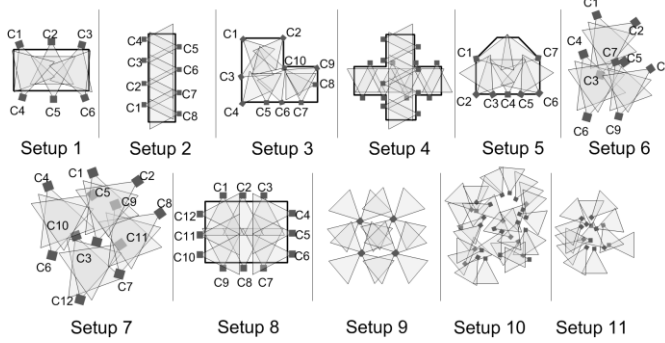|	w_E	w_R	w_F	w_D
Baseline	0	0	0.5	0.5
Initial Set 1	0.15	0.15	0.35	0.35
Initial Set 2	0.25	0.25	0.25	0.25
Initial Set 3	0	0	0.5	0.5

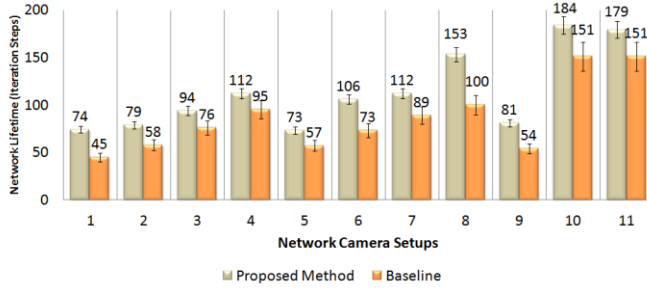Fig. 2. Examples of the different network camera setups used in simulations.

Fig. 3. Average lifetime of network for different network setups and scenarios for proposed methodology and baseline model.
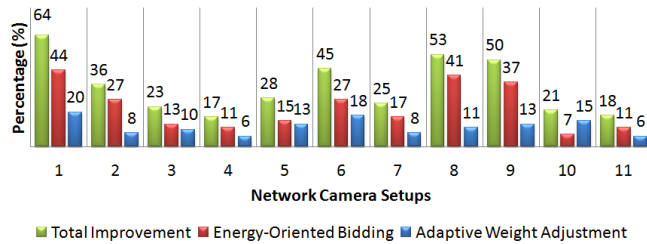

Fig. 4. Percentage of average lifetime improvement of proposed approach over baseline. Breakdown of the impact of each component.

most cameras or limited overlap (e.g. Setups 4 and 11 in Fig. 2) result in a small lifetime increase since the cameras have limited interaction and the opportunities to allocate tasks in different ways in order to conserve energy are also limited. However, in real-world environments the placement of cameras in SCNs follows some structure such that the camera FoVs will often overlap to cover an area from different angles and hence the energy gains are expected to be higher. In our experiments we found that by weighing the energy and resource factor more at the initial stage resulted in reduced lifetime increase. We have also experimented with the different energy threshold values for the adaptive weight strategy outlined in Section III.D. The best thresholds were selected experimentally from various runs, with different target placements and movements, however, we will be exploring ways to automatically compute and adjust them as

part of our future work. The best thresholds that maximized the lifetime while at the same time also consider the task requirements were between $60 - 70$ for θ_1^E and $15 - 25$ for θ_2^E. Setting the thresholds higher was found to cause unnecessary task reassignments between nodes resulting in energy lost due to communication and also from cameras executing more tasks than was otherwise necessary. Setting lower thresholds resulted in shorter lifetime improvement since the energy saving strategy was not initiated in time to migrate tasks to other cameras. The average lifetime of each network setup for the different task allocation approaches after multiple runs with different task configurations is shown in Fig. 3. For all setups, the average network lifetime is increased between $17\% - 64\%$ as shown in Fig. 4, which also illustrates the impact of each mechanism in the overall improvement, which shows that the adaptive weight strategy is a significant factor in the overall lifetime increase.

V. Conclusions

This paper presented results on distributed adaptive energy-oriented multi-task-allocation in SCNs. As an immediate follow up we aim to futher improve the adaptive strategy assignment by exploring how the different parameters can be optimally and dynamically adjusted during network operation. We will also study the impact of different objective functions on the assignment problem.

References

[1] H. Aghajan and A. Cavallaro, "*Multi-Camera Networks: Principles and Applications*" Academic Press, 2009.
[2] M. Bramberger, A. Doblander, A. Maier, B. Rinner, H. Schwabach, "Distributed embedded smart cameras for surveillance applications," *Computer* , vol.39, no.2, pp.68,75, Feb. 2006.
[3] A.T. Kamal, J.A. Farrell, A.K. Roy-Chowdhury, "Information Weighted Consensus Filters and Their Application in Distributed Camera Networks," *IEEE Transactions on Automatic Control,* vol.58, no.12, pp.3112,3125, Dec. 2013.
[4] J.C. SanMiguel, C. Micheloni, K. Shoop, G.L. Foresti, A. Cavallaro, "Self-Reconfigurable Smart Camera Networks," *Computer* , vol. 47, no. 5, pp .67-73, 2014.
[5] M. Imran, et al "Complexity Analysis of Vision Functions for Comparison of Wireless Smart Cameras," International Journal of Distributed Sensor Networks, 15 pages, 2014.
[6] B. Dieber, C. Micheloni, B. Rinner, "Resource-Aware Coverage and Task Assignment in Visual Sensor Networks," *IEEE Transactions on Circuits and Systems for Video Technology,* vol.21, no.10, pp.1424,1437, Oct. 2011.
[7] A. Cenedese, F. Cerruti, M. Fabbro, C. Masiero, L. Schenato, "Decentralized task assignment in camera networks," *49th IEEE Conference on Decision and Control (CDC)*, pp.126,131, 15-17 Dec. 2010.
[8] B. Dieber, L. Esterle, B. Rinner, "Distributed resource-aware task assignment for complex monitoring scenarios in Visual Sensor Networks," *Sixth International Conference on Distributed Smart Cameras,* pp.1,6, Oct. 30 2012-Nov. 2 2012.
[9] B. Rinner, B. Dieber, L. Esterle, P.R. Lewis, X. Yao, "Resource-aware configuration in smart camera networks," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pp.58,65, 16-21 June 2012.
[10] C. Ye, Y. Zheng, S. Velipasalar, M.C. Gursoy, "Energy-aware and robust task (re)assignment in embedded smart camera networks," *10th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS),* pp.123,128, 27-30 Aug. 2013.
[11] M. Casares, A. Pinto, Youlu Wang; S. Velipasalar, "Power consumption and performance analysis of object tracking and event detection with wireless embedded smart cameras," *3rd International Conference on Signal Processing and Communication Systems,* pp.1,8, 28-30 Sept. 2009.
[12] M.M. Zavlanos, L. Spesivtsev, G.J. Pappas, "A distributed auction algorithm for the assignment problem," *47th IEEE Conference on Decision and Control,* pp.1212, 9-11 Dec. 2008.
[13] A. T. Zimmerman, J. P. Lynch, and F. T. Ferrese. "Market-based resource allocation for distributed data processing in wireless sensor networks". *ACM Trans. Embed. Comput. Syst.* 12, 3, April 2013.
[14] G. Anastasi, M. Conti, M.Di Francesco, and A. Passarella, "Energy conservation in wireless sensor networks: A survey". *Ad Hoc Netw.* 7, 3, pp 537-568, May 2009.