Chapter 13
Self-aware Object Tracking in Multi-Camera Networks

Lukas Esterle, Jennifer Simonjan, Georg Nebehay, Roman Pflugfelder, Gustavo Fernández Domínguez, and Bernhard Rinner

Abstract This chapter discusses another example of self-aware and self-expressive systems: a multi-camera network for object tracking. It provides a detailed description of how the concepts of self-awareness and self-expression can be implemented in a real network of smart cameras. In contrast to traditional cameras, smart cameras are able to perform image analysis on-board and collaborate with other cameras in order to analyse the dynamic behaviour of objects in partly unknown environments. Self-aware and self-expressive smart cameras are even able to reason about their current state and to adapt their algorithms in response to changes in their environment and the network. Self-awareness and self-expression allow them to manage the trade-off among performance, flexibility, resources and reliability during runtime. Due to the uncertainties and dynamics in the network a fixed configuration of the cameras is infeasible. We adopt the concepts of self-awareness and self-expression for autonomous monitoring of the state and progress of each camera in the network and adapt its behaviour to changing conditions. In this chapter we focus on describing the building blocks for self-aware camera networks and demonstrate the key characteristics in a multi-camera object tracking application both in simulation and in a real camera network. The proposed application implements the goal sharing

Lukas Esterle
Alpen-Adria-Universität Klagenfurt, Austria, e-mail: lukas.esterle@aau.at

Jennifer Simonjan
Alpen-Adria-Universität Klagenfurt, Austria, e-mail: jennifer.simonjan@aau.at

Georg Nebehay
Austrian Institute of Technology, Austria, e-mail: gnebehay@gmail.com

Roman Pflugfelder
Austrian Institute of Technology, Austria, e-mail: roman.pflugfelder@ait.ac.at

Gustavo Fernández Domínguez
Austrian Institute of Technology, Austria, e-mail: gustavo.fernandez@ait.ac.at

Bernhard Rinner
Alpen-Adria-Universität Klagenfurt, Austria, e-mail: bernhard.rinner@aau.at

This chapter is part of the book: P.R. Lewis, M. Platzner, B. Rinner, J. Tørresen, X. Yao (Eds.) Self-aware Computing Systems - An Engineering Approach. Springer 2016.
pp 261-277. http://link.springer.com/chapter/10.1007/978-3-319-39675-0_13

with time-awareness capability pattern, including meta-self-awareness capabilities as discussed in Chapter 5. Furthermore, the distributed camera network employs the middleware system described in Chapter 11 to facilitate distributed coordination of tracking responsibilities. Moreover, the application uses socially inspired techniques and mechanisms discussed in Chapter 7.

13.1 Smart Camera Networks

Recent advances in technology make cameras almost omnipresent in our everyday life. Cameras are widely used for applications in security, disaster response, environmental monitoring and smart environments, among others. *Smart cameras* have emerged recently by bringing together advances in computer vision, embedded computing, image sensors and networks [415, 341]. They provide image sensing, processing, storage and communication capabilities onboard an embedded device. Smart cameras gained acceptance due to various reasons, including the low system costs, the ability to avoid network loads and the wide range of possible application scenarios. Soon enough, single smart cameras were connected to distributed smart camera networks. They are real-time, distributed, embedded systems that perform computer vision tasks using multiple cameras [45, 342, 336]. Compared to a network of traditional cameras, smart cameras offer the benefit that raw data do not have to be transmitted via the network. These raw data are processed on the sensor platform and necessary results are transmitted. As each camera has reasonable computing and communication capabilities, such a network of smart cameras can be treated as a distributed system for image processing.

In order to allow useful in-network processing of captured imagery, smart camera networks have to deal with various challenges. These challenges vary from highly dynamic behaviour of objects over partially unknown environments to required cooperation with neighbouring cameras on demand. To enable future cameras to deal with these challenges without human interaction, they are required to achieve advanced levels of autonomous behaviour to adapt themselves at runtime and learn appropriate behaviours for changing conditions. A particular challenge is to manage the trade-off of conflicting objectives such as high performance, low resource consumption and high reliability. Not knowing possible changes due to dynamics in the objects' behaviour, the environment or the network itself does not allow a fixed configuration [348]. An adaptive approach only allows a system to change based on predefined options and again only based on expected and foreseen trade-offs.

In this chapter we adopt the concepts of *self-awareness* and *self-expression* as a successful alternative to fixed configurations. We translate these concepts to computational analogies and apply them to smart camera networks [340]. As introduced in Chapter 12, self-awareness refers to the ability of a system to obtain and maintain knowledge about its state, behaviour and progress, enabling self-expression, the generation of autonomous behaviour based on such self-awareness. Combining both self-awareness and self-expression allows for adaptation of a camera's behaviour to changing conditions in an effective and autonomous manner. We therefore implement the reference architecture using the goal sharing pattern. We enhance this pattern with meta-self-aware capabilities to improve our application even further. The fundamental building blocks to achieve self-awareness and self-expression in a camera are effective sensing of the environment, learning models of the camera's state and context during runtime as well as decentralised decision making. The building blocks and their interactions are explained in this chapter in order to build a computationally self-aware and self-expressive camera network. Its features and capabilities are demonstrated with a distributed multi-camera tracking application as an example.

13.2 Object Tracking

Object tracking is an important topic and an extensively investigated subject within the field of computer vision. Research of object tracking algorithms has generated great interest in the computer vision community due to the many fields of application such as automated surveillance, security, object indexing and retrieval, human computer interaction, transportation and activity recognition. Object tracking can be classified as an intermediate-level computer vision task. Low-level information, such as edge segments or corner points, is used to build the desired trajectory in order to provide it for high-level tasks such as object retrieval or activity recognition. Basically, the goal of a tracking task is to recover the motion paths or trajectories of objects using detected object locations, such that each recovered trajectory represents the motion of a single object. Given an object i in frame t (noted as O_i^t) and a set of 'n' candidate objects in frame $(t+1)$ O_k^{t+1} (where $k = 1,...,n$), the tracking problem consists of selecting an object j in frame $(t+1)$, i.e., O_j^{t+1}, from among all 'n' objects which best matches with object O_i^t. The term object refers to image objects and the best matching is specified by some distance measure.

Unfortunately, the captured visual data is usually contaminated with noise, and missing observations increase the complexity of the tracking task. Objects can appear in different orientations, rotations and shapes depending on how they are oriented towards the camera. Occlusions can occur any time while reliable detections are still needed. Changing lighting conditions may appear in many environments and add another challenge to the tracking algorithms. Therefore, advanced techniques for data association and state estimation are necessary to provide robustness in the generation of the objects' trajectories, i.e., to succeed in the tracking task. Tracking algorithms can be classified according to different criteria, including the type of information the algorithms extract and use, the extraction method, and the matching approach (e.g., deterministic or stochastic). We refer the interested reader to [421] for a survey of tracking algorithms, to [392] for video tracking and to [184] for visual surveillance tracking.

In a multi-camera network the goal is to detect, localise and track moving objects such as pedestrians or vehicles within the fields of view (FOVs) of all cameras throughout the whole network. Besides the aforementioned difficulties, multi-camera tracking poses additional challenges due to dynamics of the environment, uncertainties of camera pose and network topology. In comparison to tracking objects in a single camera, multi-camera tracking requires the cooperation of the cameras to delegate tracking responsibilities. The use of epipolar geometry to fuse the object locations is useful when cameras have overlapping FOVs [209, 319]. However, in many scenarios and applications, cameras do not have overlapping FOVs. In such situations, assumptions about the path followed by the object or its speed [185, 214, 200], use of a motion model [320], assumptions about geometry of the scene [253] or a combination of learning the spatial links between cameras, movement of the object and colour information [298] are useful to track objects successfully.

13.3 Multi-camera Tracking Coordination

Transferring tracking from single cameras to a network of multiple cameras requires coordination of the tracking responsibility. Coordinating this responsibility for tracking an object among multiple cameras is a fundamental issue in online multi-camera tracking. A particular challenge is maintaining the association of objects when they move among cameras, i.e., to re-identify tracked objects among multiple cameras [381]. Once a desired object to be tracked is identified, the camera network has to decide by itself how to track this object through the network whenever this object is present in the observed scenario.

In centralised coordination, the cameras send the traces of the objects within their FOV to a central node which then selects the "best" trace. Various approaches have been proposed to coordinate tracking responsibilities in camera networks [199, 77, 245]. A central component for coordination of tracking responsibilities introduces benefits and drawbacks. On the downside, gathering all information on a single entity adds significant communication overhead and computational load on a single component. Furthermore, a centralised approach limits scalability and introduces a single point of failure, reducing the applicability of this approach in large camera networks. On the upside, a centralised approach may achieve a better tracking performance due to the availability of complete tracking and state information from all sensors in a single node. A comprehensive analysis of the state of the art is given in [322, 280].

In distributed coordination each camera decides on its own when and to whom to hand over the tracking responsibility. This distributes the computational load to the cameras in the network and reduces the communication overhead by avoiding transmitting full state information of all cameras. This makes the network not only highly scalable but also quite robust to failure of single cameras or even to changes in network topology caused by dynamically adding cameras. Various approaches for distributed coordination without centralised control have been presented in the literature [118, 328, 128]. Deriving the handover decision based on possibly incomplete, local information and with a camera's limited resources is still a fundamental challenge in distributed coordination.

In the following we summarise how self-aware and self-expressive approaches are used in order to enable a network of smart cameras to coordinate tracking responsibilities autonomously and efficiently among each other.

13.4 Self-aware and Self-expressive Building Blocks

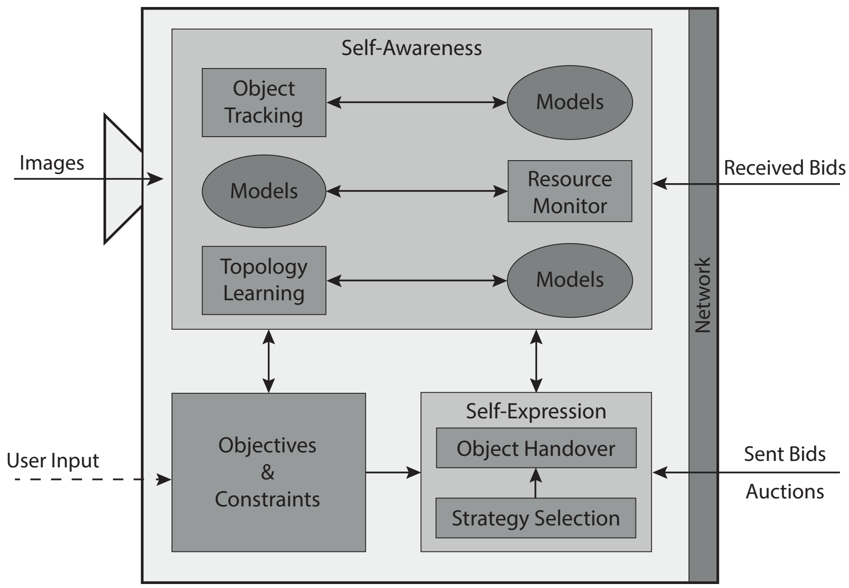


Fig. 13.1 The architecture of a self-aware and self-expressive camera node composed by six building blocks

To endow a camera network with self-awareness and self-expression, dedicated self-aware and self-expressive building blocks are implemented in the individual camera nodes. The different building blocks are able to interact with blocks locally on each camera as well as with other blocks in the network on remote cameras. The blocks depicted in Figure 13.1 aim for a generic and flexible design and implement architectural self-aware and self-expressive patterns as discussed in Chapter 5. Self-aware building blocks, such as *object tracking*, *resource monitoring*, and *topology learning*, are able to monitor its state and behaviour. Utilising online learning techniques allows them to maintain models of their states and the respective behaviour rather than rely on predefined knowledge and rules. The generated models are then used in the self-expressive building blocks (i.e., *object handover* and *strategy selection*) to steer the behaviour of the entire system. The *objectives & constraints* block represents the camera's goals and resource constraints. Both highly influence the other blocks and hence the behaviour and interaction of each individual camera in the network.

The aggregation of individual camera nodes allows the composition of a truly self-aware and self-expressive decentralised camera network. As our employed embedded camera platforms are rather resource-constrained, we focussed the design of each individual building block on resource awareness. Thus, while each building block can be utilising a diverse number of algorithms from computer vision, online learning, distributed coordination and decision making for their implementation, all building blocks have to be able to execute in real time.

In an iterative design process, we initially implemented the interaction awareness pattern in order to allow the cameras to learn about their neighbouring cameras. This allowed the individual cameras to coordinate tracking responsibilities among local neighbouring cameras rather than the entire network. In the next step, we introduced time awareness and enabled each individual camera to also "unlearn" previous learnt information. This becomes important in case the network changes during runtime. In the final step for this application, we introduced meta-self-aware capabilities. These capabilities allow the cameras to trade off exploration, identifying local changes in the network, and exploitation, using the previously learnt information in order to optimise coordination of tracking responsibilities.

13.4.1 Object Tracking

The *object tracking* (OT) building block of each camera is responsible for acquiring images, detecting objects and tracking them within the camera's FOV. Additionally, the OT block transmits images and tracking results to other interested components in the system (for example, the user interface), and if necessary it can update the model of the object during runtime. Identifying the model within the FOV of a camera relates to *private self-awareness* whereas creating and adapting the model of an object corresponds to the *model* in our reference architecture (cf. Chapter 3). The tracking process is described as semi-automatic because the user has to select the desired object to be tracked. After initialising the process, the computer vision tasks run automatically.

The implementation of this self-aware and self-expressive camera network application has been performed in multiple iterations. While the initial version was limited to tracking a single object within the camera network, the final version was capable of simultaneously tracking three people on a network of six smart cameras. The camera network employs appearance-based tracking without using temporal information from previous frames. As soon as the object has been identified within the FOV of the camera, the OT starts tracking the object. By doing so, the OT can adapt the visual representation of the object and improve the internal model if this is desired. This simple approach achieves the necessary computational resource and real-time requirements as well as acceptable accuracy and robustness against dropped frames, occlusions and disappearance of objects.

The approach employs general assumptions about the cameras and visual discriminability of objects. The final implementation of the camera network [362] exploited the static camera assumption and built a detailed model of the background. The static background model enables us to implement a fast and reliable foreground object detector. To perform the association of foreground objects with the desired object to be tracked, colour histograms are compared using appropriate distance metrics. The method is illustrated in Figure 13.2. In a first step, foreground pixels in each camera are identified by comparing the camera image to the background image learned by each camera individually. Foreground constitutes moving objects that comprise objects of interest for tracking. These foreground pixels are then grouped into candidate objects based on their connectedness. In a second step, an association is performed between these candidate objects and a template database that contains the objects of interest (Figure 13.2(d)). To this end, a measure of similarity is employed according to [82] which is interpreted as the confidence of the validity of the association. This is depicted by different colours in Figure 13.2(c).

The association between templates and candidates is established by interpreting the problem of associating templates and candidates as a transportation problem, where the distances between the respective feature vectors are the transportation costs and the goal is to minimise all transportation costs. This problem can be solved optimally by employing the well-known Hungarian algorithm [227]. Additionally, the reciprocal of the transportation cost for a successful association is reported to other components as a confidence value of the current object. In each frame, the *object tracking* block searches for an assignment minimising the overall transportation cost of the system. In this way, a satisfying tracking performance is achieved even in difficult scenarios.

13.4.2 Object Handover

The *object handover* block coordinates the object tracking responsibility in the camera network. We apply a novel market-based handover approach [125]. A more detailed description of this approach is presented in Section 7.4. In this artificial market, the cameras act as traders and treat object tracking responsibilities as goods. For trading purposes, an artificial currency is used. This currency is provided by each object tracking responsibility as some utility over time. This makes these responsibilities worthwhile for cameras to own. The cameras can decide in a self-expressive manner on their own when to "sell" tracking responsibilities to other cameras using single sealed-bid auctions. Employing the Vickrey auction mechanism, which sells the good to the highest bidder for the second highest price, makes truthful bidding the dominant strategy among the participating cameras. Whenever a camera decides to sell an object, it initiates an auction for this particular object by transferring an

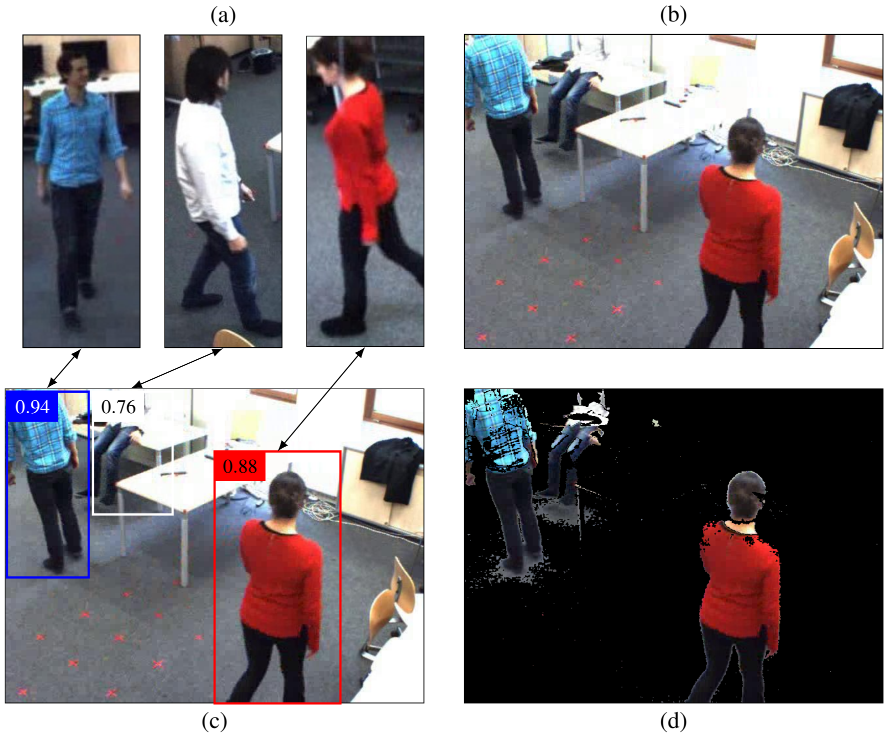


Fig. 13.2 Object tracking approach. (a) Template database images, (b) original image, (c) matching results between foreground objects and objects templates, (d) foreground image

object description to the other cameras. The receiving cameras request the state of the searched object from the *object tracking* block. This means, the *object tracking* block searches within its own FOV for the object and values the object based on detection confidence and visibility. The *object handover* blocks of the cameras return their valuation as a bid. The auctioneering camera selects the highest bidder and transfers the tracking responsibility. An important question for the selling camera is to whom to send the auction invitations. Without any a priori knowledge about the network topology, invitations could be broadcasted to all cameras. By following this policy the "best" camera for taking over the tracking responsibility will receive an invitation (and may respond with the highest bid). However, the BROADCAST policy causes a significant communication and computation load because each camera has to perform an object detection after receiving the invitation.

13.4.3 Topology Learning

If the auctioneering cameras are aware of the potentially "best" cameras in their neighbourhood, this knowledge can be exploited to significantly reduce the overhead. Such topological information can be initially assigned to the cameras or computed by means of multi-camera calibration during the deployment of the camera network. However, in a self-aware manner we learn the network topology by observing the bidding behaviour of cameras over time. Each camera keeps track of its local neighbours and uses artificial pheromones to express the likelihood of a handover to that camera. Whenever a handover has taken place, the artificial pheromone to the succeeding camera is strengthened. If no trading occurs, pheromones evaporate over time. This mechanism enables each camera to deal with network uncertainties and to adapt to changes in its neighbourhood topology caused by addition, removal or failure of cameras or changes in the movement pattern of the objects.

We exploit the learnt neighbourhood topology through three different communication policies for the handover: broadcast auctions to all cameras (BROADCAST), a smooth probabilistic multicast (SMOOTH) and a threshold-based probabilistic multicast (STEP) [125]. The SMOOTH policy sends auction invitations to all neighbours, with probability normalised to the current pheromone level. The STEP policy sends invitations to all neighbours with pheromone level above a certain threshold and to neighbours below the threshold with some (low) probability. Details and formal definitions to these different policies are also given in Section 7.4.1.1. Additionally, it is not only important to whom to advertise object tracking responsibilities, but also when to initiate such auctions. We distinguish between sending out invitations at regular intervals (ACTIVE) or only when the object is about to leave the FOV (PASSIVE). While a PASSIVE schedule ensures we keep track of each object continuously, in comparison the ACTIVE schedule achieves higher network-wide tracking utility as cameras assign tracking responsibilities to the camera "seeing" the object best at all times. Combining the variations in which cameras to invite and when to send out the invitations results in six different self-expressive handover strategies, each of which obviously influences tracking utility as well as communication and computational overhead.

13.4.4 Strategy Selection

While the six handover strategies allow us trade off communication overhead for tracking utility and hence influence the behaviour of the network, selecting a strategy is a difficult decision. The performance of each strategy strongly depends on factors such as the placement of the cameras, the movement patterns of the objects and the object tracking algorithm. In principle, we can follow three approaches for strategy selection: (i) a homogeneous assignment, where all cameras employ the same strategy from deployment time on, (ii) a heterogeneous assignment, where at least two cameras in the network use different strategies from deployment time on, and

(iii) a dynamic selection, where each camera can select its strategy autonomously during runtime.

As discussed in Section 7.4.1.3, we use online learning algorithms, specifically multi-armed bandit problem solvers, within each camera to learn the appropriate strategy for each node during runtime to trade off communication overhead for achieved tracking utility. The bandit solvers balance the exploitation behaviour, where a camera achieves high performance by using its currently best known strategy, with exploration, where the camera explores the effect of using other strategies to build up its knowledge [239, 238]. Dynamic strategy selection leads to a meta-self-aware behaviour of the individual camera nodes and by extension of the entire camera network. This allows the network as a whole to achieve a more Pareto efficient global performance than with any static strategy assignment at deployment time.

13.4.5 Resource Monitoring

Resource monitoring is an important aspect of computational self-awareness, and its main objective is to observe the available resources on the camera nodes. The monitored data is further used to build up models of resource consumption for each task a camera is capable of performing. The knowledge generated by the self-aware *resource monitoring* block is provided to the individual self-aware and self-expressive blocks on the camera. *Object handover* can use this information on the one hand to reason about submitting bids for a new object tracking responsibility and on the other hand to factor in available resources at the time of bidding for its valuation of the object. The *strategy selection* block uses the information from the resource monitor to reason not only about the performance of each task but also about its respective resource consumption. In our network, we currently monitor required processing power, available and allocated memory, and network traffic.

13.4.6 Constraints and Objectives

In order for a camera to become self-aware, it not only requires constraints, objectives and goals but also has to be aware of them. In our system every camera has its own constraints and objectives which it needs to consider for its self-aware and self-expressive operation. Constraints specify some limitations of the available resources (processing, memory and networking). These constraints help us decide on whether to bid for an object, but also help us evaluate the performance of the different strategies. In contrast to hardware-defined constraints, objectives are defined by the user during runtime or the designer of the system before deployment. Objects in our system drive the behaviour of the cameras and specify, for example, some quality of service parameters or certain tasks the cameras should achieve.

13.5 Camera Network Case Study

13.5.1 Camera Network Setup

For our experimental study, we set up a smart camera network in our laboratory at Alpen-Adria-Universität Klagenfurt. The network consists of six cameras, four of them in the laboratory room with overlapping fields of view and two more in the corridor and lounge area, respectively. An illustration of the camera layout is given in Fig. 13.3. Figure 13.4 shows snapshots of the six cameras' FOVs. This heterogeneous network is composed of different hardware platforms. Cameras 1 to 4 are equipped with Atom processors and connected via wired Ethernet. Cameras 5 and 6 are based on Pandaboards equipped with ARM processors and use WiFi for communication. All cameras run standard Linux and our distributed publish-subscribe middleware system to provide a flexible software platform for software development (cf. Chapter 11). The building blocks have been implemented in C++ and C# using the middleware services for communication and control.

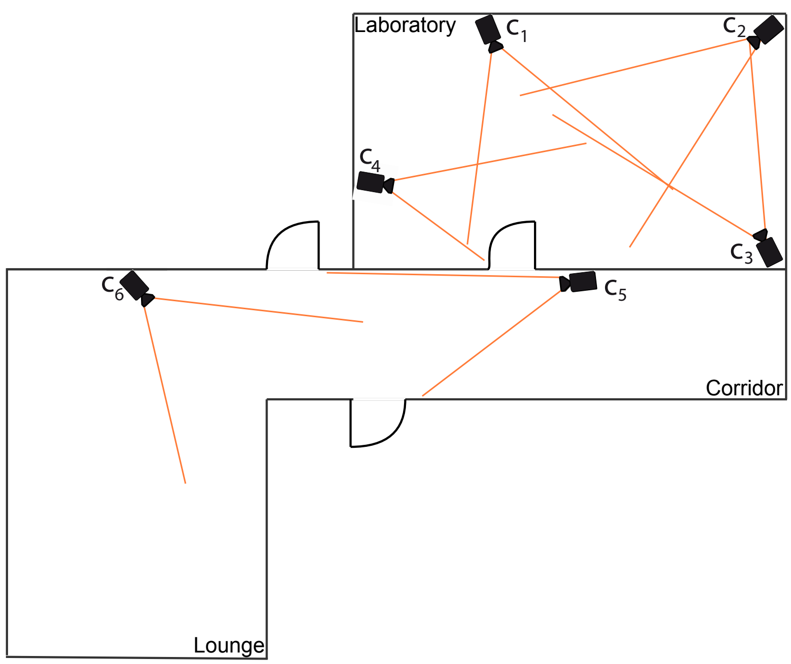
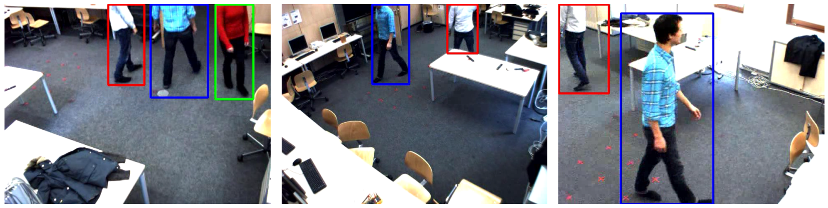
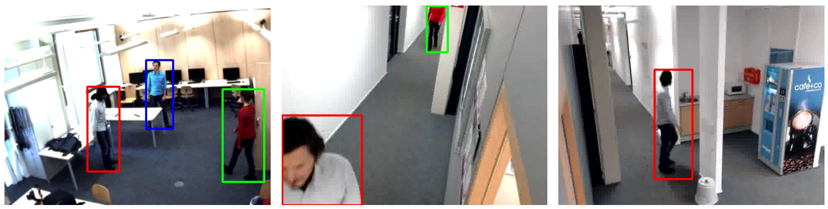


Fig. 13.3 The smart camera network composed of six cameras deployed in an indoor environment. The cameras are depicted by a black camera symbol and their FOVs are indicated by orange lines.



(a) Camera 1, frame nr. 2069 (b) Camera 2, frame nr. 688 (c) Camera 3, frame nr. 719



(d) Camera 4, frame nr. 1027 (e) Camera 5, frame nr. 2593 (f) Camera 6, frame nr. 1760

Fig. 13.4 Snapshots (a) to (f) of cameras 1–6 at different times. The images show the status of object tracking over time; the system tracks three people who are marked by red, blue and green bounding boxes, respectively.

13.5.2 Tracking Results

For the evaluation of our object tracking block we use the smart camera network with people walking through the indoor environment (as depicted in Fig. 13.4). The performance metrics used in the evaluation process are based on state-of-the-art metrics in the area of object tracking and multi-camera systems [31, 431, 422]. Detection of people and people tracking are evaluated per camera and across all cameras. Performance metrics such as sensitivity, precision and accuracy are used in the case of object detection evaluation. High-level metrics such as correct detected track (CDT), track detection failure (TDF) and false alarm track (FAT) show an overall view of performance of the tracking system.

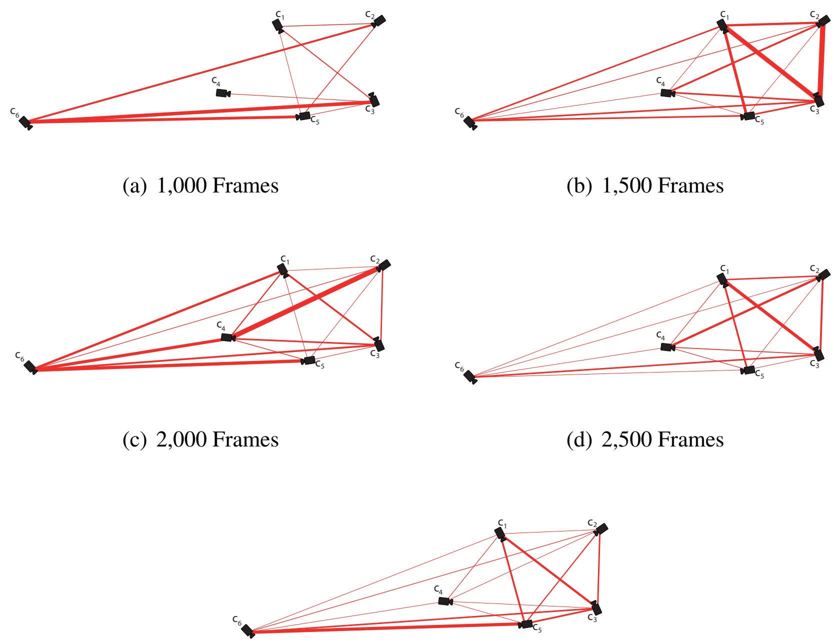
Tables 13.1 and 13.2 summarise the results of a typical scenario of concurrently tracking three selected persons independently walking around in the indoor environment for around 120 seconds. While the former shows for each camera the sensitivity and CDT for object detection and object tracking, respectively, the latter summarises the percentage of correct objects tracked between cameras. In this scenario, the tracked persons were not continuously visible to all cameras; at some points, participants were not seen by any camera at all. The cameras mounted in the laboratory (cameras 1–4) achieved better detection and tracking results compared to the performance of cameras 5 and 6. Here the tracking performance degraded slightly due to changes in lighting and object appearance.

Table 13.1 Performance of object detection and object tracking: single camera evaluation

Camera number	Object detection: Sensitivity	Object tracking: CDT
1	0.78	3
2	0.73	2
3	0.87	3
4	0.76	3
5	0.48	2
6	0.54	2

Table 13.2 Performance of correctly tracked objects between cameras

Camera pair	(1,2)	(1,3)	(1,4)	(1,5)	(2,3)	(2,4)	(2,5)	(3,4)	(3,6)	(4,5)	(5,6)
Percentage	66.67	100.00	100.00	66.67	66.67	66.67	33.33	100.00	66.67	66.67	66.67



(a) 1,000 Frames

(b) 1,500 Frames

(c) 2,000 Frames

(d) 2,500 Frames

(e) 3,000 Frames

Fig. 13.5 The graph shows the learnt topology at various times of the experiment exploiting the trading behaviour. The thickness of the red line indicates the pheromone level of the link.

13.5.3 Topology Learning

Our self-aware topology learning block builds up a neighbourhood relationship graph locally on each camera. Figure 13.5 shows snapshots of the graph for the entire network at various time steps in a typical test run. The thickness of the red lines indicates the strength of the artificial pheromone deposit on this link, and corresponds to the probability of an object transiting between the connected cameras. Initially, links are created between the camera in the lounge (camera 6) and those in the laboratory (cameras 1–4) due to misdetections of camera 5. Nevertheless, due to the evaporation of the artificial pheromones, these inaccurate links are "forgotten" over time. Furthermore, cameras can deal not only with errors induced by the tracking block but also with changes in the topology due to hardware errors, vandalism, or maintenance when cameras are being removed, added, or moved to a different location. Over time invalid links evaporate and a qualitatively correct neighbourhood graph emerges again.

13.5.4 Communication and Utility Trade-off

We evaluated the effect of the handover strategy on the overall tracking utility and communication overhead. Figure 13.6 depicts this trade-off between utility and communication for homogeneous strategy selection in our smart camera network. The utility is defined as the aggregated tracking utility of all cameras, and the communication is defined as the number of all sent auction messages during the entire tracking operation. Both utility and communication values are normalised by those from the best strategy (i.e., ACTIVE BROADCAST). The six strategies result in six different trade-offs for utility and communication. An operator overseeing the network can select a strategy based on the current situation and needs. These requirements may vary for example when the attention is directed from general surveillance to tracking a single person.

In addition, we also analysed homogeneous and heterogeneous strategy assignments, as well as dynamic strategy selection during runtime and their achieved trade-off in our CamSim simulation tool[1] [123]. Obviously, heterogeneous selection (black crosses) leads to many more outcomes in the objective space. The extension of the Pareto efficient frontier brought about by heterogeneity in comparison to the results by homogeneity is also apparent. However, it is also clear that the outcomes of many heterogeneous strategies are dominated, and many are strictly worse than the original outcomes from the homogeneous strategies. As an operator deploys networks in partially unknown environments and cannot foresee the dynamic behaviour of the objects, an optimal heterogeneous selection is impossible. This clearly benefits the self-expressive behaviour of our cameras, facilitating a dynamic strategy

[1] http://www.epics-project.eu/CamSim/

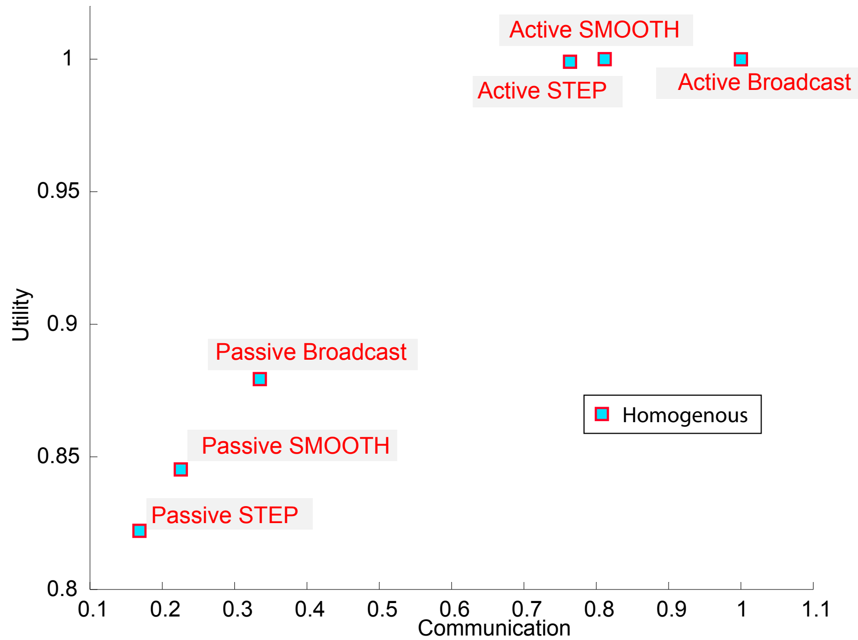


Fig. 13.6 Performance for two exemplary scenarios from our smart camera network showing homogeneous (red and blue squares), heterogeneous (black crosses), and dynamically learned strategy assignments (coloured symbols). The results have been normalised by the maximum value of the ACTIVE BROADCAST strategy and are averages over 30 runs with 1,000 time steps each [239, 238].

selection. Dynamic strategy selection (coloured symbols) is able to outperform the static (homogeneous and heterogeneous) strategies and to extend the Pareto front.

13.6 Conclusion and Outlook

This chapter presented how self-awareness and self-expression were implemented in a real smart camera network for a person tracking application. By enabling each camera to learn about its environment, its topology, and its performance, the entire network was able to perform continuously well to achieve a common goal. The facilitated building blocks allowed the camera network to face various challenges such as the limitation of available resources of the cameras, the continuous changes in a real scenario and the intrinsic problem of robustness in people tracking. These six different building blocks, encapsulating the entire processing, were embedded into resource-limited smart camera nodes and aggregated into a completely decentralised and thus scalable network.

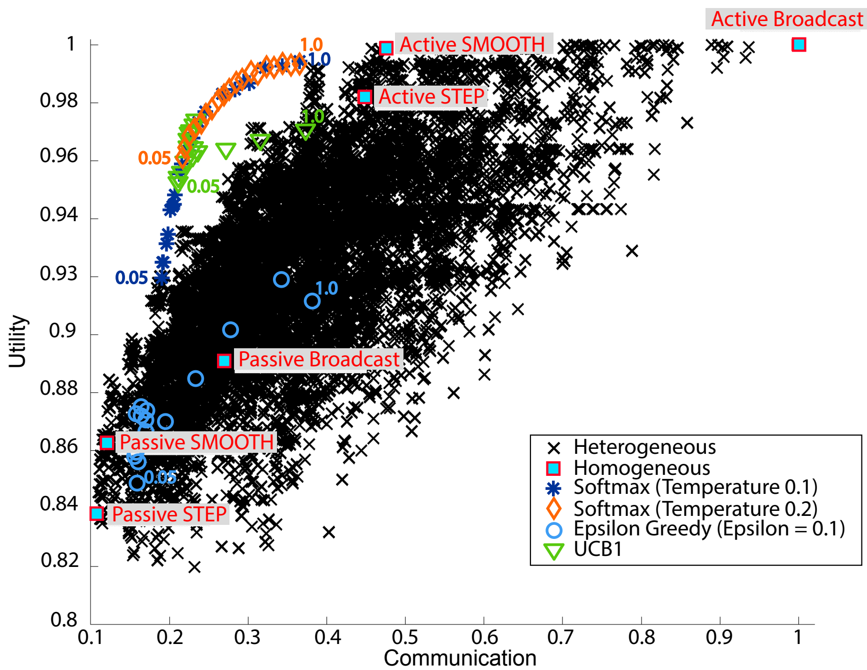


Fig. 13.7 Performance for two exemplary scenarios from our simulation environment showing homogeneous (red and blue squares), heterogeneous (black crosses), and dynamically learned strategy assignments (coloured symbols). The results have been normalised by the maximum value of the ACTIVE BROADCAST strategy and are averages over 30 runs with 1,000 time steps each [239, 238].

Nevertheless, the different blocks are implemented at different levels of self-awareness and self-expression. While the *object tracking* block is only *stimulus-aware*, other blocks such as the *object handover*, *topology learning*, or *strategy selection* are *interaction-aware*, *time-aware*, and *meta-self-aware*, respectively. Merging the different blocks in every single camera allows each camera, and by extension the entire network, to achieve higher levels of self-awareness and self-expression. However, one could still introduce additional blocks or refine existing ones in order to improve the overall performance. An example would be the *object handover* block which currently combines multiple strategies, each one consisting of an auction schedule and a communication policy. In the presented version, the auction schedules are static for all cameras, and communicate either at regular intervals or when the object is at a specified position within the FOV of the camera. In contrast, a camera could learn the best timing for a handover during runtime. In such a setting, the camera could start with an active approach and refine the timing based on the received bid in the advertised auctions.

The concepts of computational self-awareness and self-expression are not limited to camera networks alone. The previous chapter presented another application and the next chapter introduces a third one using real-time interaction between humans playing music. In fact, we are confident that self-awareness and self-expression could serve as an enabling technology for future systems and networks, meeting a multitude of requirements with respect to functionality, flexibility, performance, resource usage, costs, reliability and safety.

Acknowledgements

The research leading to these results was conducted in the EPiCS project (Engineering Proprioception in Computing Systems) and received funding from the European Union Seventh Framework Programme under grant agreement no. 257906.

The contributors would like to acknowledge additional financial support for research performed in individual chapters of this book.

- Chapters 6 and 7 were also supported by EPSRC Grants (Nos. EP/I010297/1, EP/K001523/1 and EP/J017515/1).
- Chapter 8 was also supported by the German Research Foundation (DFG) within the Collaborative Research Centre "On-The-Fly Computing" (SFB 901) and the International Graduate School on Dynamic Intelligent Systems of Paderborn University.
- Chapter 9 was also supported in part by HiPEAC NoE, by the European Union Seventh Framework Programme under grant agreement numbers 287804 and 318521, by the UK EPSRC, by the Maxeler University Programme, and by Xilinx.
- Chapter 12 was also supported in part by the China Scholarship Council, by the European Union Seventh Framework Programme under grant agreement numbers 287804 and 318521, by the UK EPSRC, by the Maxeler University Programme, and by Xilinx.
- Chapter 13 was also supported by the research initiative Mobile Vision with funding from the Austrian Institute of Technology and the Austrian Federal Ministry of Science, Research and Economy HRSMV programme BGBl. II no. 292/2012.
- Chapter 14 was also supported by the Research Council of Norway under grant agreement number 240862/F20.
- Peter Lewis would like to thank the participants of the Dagstuhl Seminar "Model-Driven Algorithms and Architectures for Self-aware Computing Systems", Seminar Number 15041, for many insightful discussions on notions of self-aware computing.

References

1. Aberdeen, D., Baxter, J.: Emmerald: a fast matrix-matrix multiply using Intel's SSE instructions. Concurrency and Computation: Practice and Experience **13**(2), 103–119 (2001)
2. Abramowitz, M., Stegun, I.: Handbook of Mathematical Functions. Dover Publications (1965)
3. Agarwal, A., Harrod, B.: Organic computing. Tech. Rep. White paper, MIT and DARPA (2006)
4. Agarwal, A., Miller, J., Eastep, J., Wentziaff, D., Kasture, H.: Self-aware computing. Tech. Rep. AFRL-RI-RS-TR-2009-161, MIT (2009)

5. Agne, A., Hangmann, H., Happe, M., Platzner, M., Plessl, C.: Seven recipes for setting your FPGA on fire – a cookbook on heat generators. Microprocessors and Microsystems **38**(8), 911–919 (2014). DOI 10.1016/j.micpro.2013.12.001
6. Agne, A., Happe, M., Keller, A., Lübbers, E., Plattner, B., Platzner, M., Plessl, C.: ReconOS: An Operating System Approach for Reconfigurable Computing. IEEE Micro **34**(1), 60–71 (2014). DOI 10.1109/MM.2013.110
7. Agne, A., Platzner, M., Lübbers, E.: Memory virtualization for multithreaded reconfigurable hardware. In: Proceedings of the International Conference on Field Programmable Logic and Applications (FPL), pp. 185–188. IEEE Computer Society (2011). DOI 10.1109/FPL.2011.42
8. Ahuja, S., Carriero, N., Gelernter, D.: Linda and friends. IEEE Computer **19**(8), 26–34 (1986). DOI 10.1109/MC.1986.1663305
9. Al-Naeem, T., Gorton, I., Babar, M.A., Rabhi, F., Benatallah, B.: A quality-driven systematic approach for architecting distributed software applications. In: Proceedings of the 27th International Conference on Software Engineering, pp. 244–253. ACM (2005). DOI 10.1145/1062455.1062508. URL http://doi.acm.org/10.1145/1062455.1062508
10. Ali, H.A., Desouky, A.I.E., Saleh, A.I.: Studying and Analysis of a Vertical Web Page Classifier Based on Continuous Learning Naive Bayes (CLNB) Algorithm, pp. 210–254. Information Science (2009)
11. Alippi, C., Boracchi, G., Roveri, M.: Just-in-time classifiers for recurrent concepts. IEEE Transactions on Neural Networks and Learning Systems **24**(4), 620–634 (2013)
12. Amir, E., Anderson, M.L., Chaudhri, V.K.: Report on DARPA workshop on self-aware computer systems. Tech. Rep. UIUCDCS-R-2007-2810, UIUC Comp. Sci. (2007)
13. ANA: Autonomic Network Architecture. URL www.ana-project.org. (accessed March 8, 2016)
14. Angelov, P.: Nature-inspired methods for knowledge generation from data in real-time (2006). URL http://www.nisis.risk-technologies.com/popup/Mallorca2006_Papers/A333_13774_Nature-inspiredmethodsforKnowledgeGeneration_Angelov.pdf
15. Apache: Hadoop. http://hadoop.apache.org/docs/r1.2.1/mapred_tutorial.html. (Accessed March 8, 2016)
16. Araya-Polo, M., Cabezas, J., Hanzich, M., Pericàs, M., Rubio, F., Gelado, I., Shafiq, M., Morancho, E., Navarro, N., Ayguadé, E., Cela, J.M., Valero, M.: Assessing accelerator-based HPC reverse time migration. IEEE Transactions on Parallel and Distributed Systems **22**(1), 147–162 (2011)
17. Asanovic, K., Bodik, R., Catanzaro, B.C., Gebis, J.J., Husbands, P., Keutzer, K., Patterson, D.A., Plishker, W.L., Shalf, J., Williams, S.W., Yelick, K.A.: The landscape of parallel computing research: A view from Berkeley. Tech. Rep. UCB/EECS-2006-183, EECS Department, University of California, Berkeley (2006)
18. Asendorpf, J.B., Warkentin, V., Baudonnière, P.M.: Self-awareness and other-awareness. II: Mirror self-recognition, social contingency awareness, and synchronic imitation. Developmental Psychology **32**(2), 313 (1996)
19. Athan, T.W., Papalambros, P.Y.: A note on weighted criteria methods for compromise solutions in multi-objective optimization. Engineering Optimization **27**(2), 155–176 (1996)
20. Auer, P., Cesa-Bianchi, N., Fischer, P.: Finite-time analysis of the multiarmed bandit problem. Machine Learning **47**(2–3), 235–256 (2002)
21. Babaoglu, O., Binci, T., Jelasity, M., Montresor, A.: Firefly-inspired heartbeat synchronization in overlay networks. In: First International Conference on Self-Adaptive and Self-Organizing Systems (SASO), pp. 77–86 (2007)
22. Babenko, B., Yang, M.H., Belongie, S.: Robust object tracking with online multiple instance learning. IEEE Transactions on Pattern Analysis and Machine Intelligence **33**(8), 1619–1632 (2011)
23. Bader, J., Zitzler, E.: HypE: an algorithm for fast hypervolume-based many-objective optimization. Tech. Rep. TIK 286, Computer Engineering and Networks Laboratory, ETH Zurich, Zurich (2008)

24. Baena-García, M., Campo-Ávila, J.D., Fidalgo, R., Bifet, A.: Early drift detection method. In: Procedings of the 4th ECML PKDD International Workshop on Knowledge Discovery From Data Streams (IWKDDS), pp. 77–86. Berlin, Germany (2006)
25. Baker, S.: The identification of the self. Psyc. Rev. **4**(3), 272–284 (1897)
26. Banks, A., Gupta, R.: MQTT Version 3.1.1. http://docs.oasis-open.org/mqtt/mqtt/v3.1.1/os/mqtt-v3.1.1-os.html (2014)
27. Bartolini, D.B., Sironi, F., Maggio, M., Cattaneo, R., Sciuto, D., Santambrogio, M.D.: A Framework for Thermal and Performance Management. In: Proceedings of the Workshop on Managing Systems Automatically and Dynamically (MAD) (2012)
28. Basheer, I.A., Hajmeer, M.: Artificial neural networks: fundamentals, computing, design, and application. Journal of Microbiological Methods **43**(1), 3–31 (2000)
29. Basseur, M., Zitzler, E.: Handling uncertainty in indicator-based multiobjective optimization. International Journal of Computational Intelligence Research **2**(3), 255–272 (2006)
30. Basudhar, A., Dribusch, C., Lacaze, S., Missoum, S.: Constrained efficient global optimization with support vector machines. Structural and Multidisciplinary Optimization **46**(2), 201–221 (2012)
31. Baumann, A., Boltz, M., Ebling, J., Koenig, M., Loos, H.S., Merkel, M., Niem, W., Warzelhan, J.K., Yu, J.: A review and comparison of measures for automatic video surveillance systems. EURASIP Journal on Image and Video Processing **2008**(4) (2008). DOI 10.1155/2008/824726
32. Becker, T., Agne, A., Lewis, P.R., Bahsoon, R., Faniyi, F., Esterle, L., Keller, A., Chandra, A., Jensenius, A.R., Stilkerich, S.C.: EPiCS: Engineering proprioception in computing systems. In: Proceedings of the International Conference on Computational Science and Engineering (CSE), pp. 353–360. IEEE Computer Society (2012)
33. Ben-Hur, A., Weston, J.: A user's guide to support vector machines. Data Mining Techniques for the Life Sciences **609**, 223–239 (2010)
34. Betts, A., Chong, N., Donaldson, A.F., Qadeer, S., Thompson, P.: GPUVerify: a verifier for GPU kernels. In: Proceedings of the ACM International Conference on object-oriented Programming Systems Languages and Applications (OOPSLA) (2012)
35. Beume, N., Naujoks, B., Emmerich, M.: SMS-EMOA: Multiobjective selection based on dominated hypervolume. European Journal on Operational Research **181**(3), 1653–1669 (2007)
36. Bevilacqua, F., Zamborlin, B., Sypniewski, A., Schnell, N., Guédy, F., Rasamimanana, N.: Continuous realtime gesture following and recognition. In: Gesture in embodied communication and human-computer interaction, pp. 73–84. Springer (2010)
37. Biehl, J.T., Adamczyk, P.D., Bailey, B.P.: Djogger: A mobile dynamic music device. In: Proceedings of CHI '06 Extended Abstracts on Human Factors in Computing Systems, pp. 556–561. ACM (2006)
38. Bishop, C.M.: Neural Networks for Pattern Recognition. Oxford University Press, United Kingdom (2005)
39. Bojic, I., Lipic, T., Podobnik, V.: Bio-inspired clustering and data diffusion in machine social networks. In: Computational Social Networks, pp. 51–79. Springer (2012)
40. Bongard, J., Lipson, H.: Evolved machines shed light on robustness and resilience. Proceedings of the IEEE **102**(5), 899–914 (2014)
41. Bongard, J., Zykov, V., Lipson, H.: Resilient machines through continuous self-modeling. Science **314**(5802), 1118–1121 (2006)
42. Borkar, S.: Designing Reliable Systems from Unreliable Components: The Challenges of Transistor Variability and Degradation. IEEE Micro pp. 10–16 (2005)
43. Bouabene, G., Jelger, C., Tschudin, C., Schmid, S., Keller, A., May, M.: The Autonomic Network Architecture (ANA). IEEE Journal on Selected Areas in Communications **28**(1), 4–14 (2010). DOI 10.1109/JSAC.2010.100102
44. Boyd, J.: The Essence of Winning and Losing. http://dnipogo.org/john-r-boyd/ (1996). (Accessed March 8, 2016)
45. Bramberger, M., Doblander, A., Maier, A., Rinner, B., Schwabach, H.: Distributed Embedded Smart Cameras for Surveillance Applications. IEEE Computer **39**(2), 68–75 (2006)

46. Brdiczka, O., Crowley, J.L., Reignier, P.: Learning situation models in a smart home. IEEE Transactions on Systems, Man, and Cybernetics, Part B **39**, 56–63 (2009)
47. Breiman, L.: Bagging predictors. Machine Learning **24**(2), 123–140 (1996)
48. Breiman, L.: Random forests. Machine Learning **45**(1), 5–32 (2001)
49. Brockhoff, D., Zitzler, E.: Improving hypervolume-based multiobjective evolutionary algorithms by using objective reduction methods. In: Proceedings of the 2007 IEEE Congress on Evolutionary Computation, pp. 2086–2093 (2007)
50. Buchanan, J.T.: A naive approach for solving MCDM problems: The GUESS method. Journal of the Operational Research Society **48**(2), 202–206 (1997)
51. Buck, J.: Synchronous rhythmic flashing of fireflies. The Quarterly Review of Biology **13**(3), 301–314 (1938)
52. Buck, J.: Synchronous rhythmic flashing of fireflies II. The Quarterly Review of Biology **63**(3), 265–289 (1988)
53. Burke, E.K., Gendreau, M., Hyde, M., Kendall, G., Ochoa, G., Ozcan, E., Qu, R.: Hyper-heuristics: A survey of the state of the art. Journal of the Operational Research Society **206**(1), 241–264 (2013)
54. Buschmann, F., Henney, K., Douglas, S.C.: Pattern-oriented software architecture: On patterns and pattern languages. John Wiley and Sons (2007)
55. Buss, A.H.: Self-consciousness and social anxiety. W. H. Freeman, San Fransisco, CA, USA (1980)
56. Calinescu, R., Ghezzi, C., Kwiatkowska, M., Mirandola, R.: Self-adaptive software needs quantitative verication at runtime. Communications of the ACM **55**(9), 69–77 (2012)
57. Caramiaux, B., Wanderley, M.M., Bevilacqua, F.: Segmenting and parsing instrumentalists' gestures. Journal of New Music Research **41**(1), 13–29 (2012)
58. Carver, C.S., Scheier, M.: Attention and Self-Regulation: A Control-Theory Approach to Human Behavior. Springer (1981)
59. de Castro, L.N.: Fundamentals of natural computing: basic concepts, algorithms, and applications. Chapman & Hall/CRC Computer and Information Sciences (2006)
60. Chandra, A.: A methodical framework for engineering co-evolution for simulating socio-economic game playing agents. Ph.D. thesis, The University of Birmingham (2011)
61. Chandra, A., Nymoen, K., Volsund, A., Jensenius, A.R., Glette, K., Torresen, J.: Enabling participants to play rhythmic solos within a group via auctions. In: Proceedings of the International Symposium on Computer Music Modeling and Retrieval (CMMR), pp. 674–689 (2012)
62. Chandra, A., Yao, X.: Ensemble learning using multi-objective evolutionary algorithms. Journal of Mathematical Modelling and Algorithms **5**(4), 417–445 (2006)
63. Chang, C., Wawrzynek, J., Brodersen, R.W.: BEE2: a high-end reconfigurable computing system. IEEE Transactions on Design & Test of Computer **22**(2), 114–125 (2005)
64. Chen, J., John, L.K.: Efficient program scheduling for heterogeneous multi-core processors. In: Proceedings of the Design Automation Conference (DAC). ACM (2009)
65. Chen, R., Lewis, P.R., Yao, X.: Temperature management for heterogeneous multi-core FPGAs using adaptive evolutionary multi-objective approaches. In: Proceedings of the International Conference on Evolvable Systems (ICES), pp. 101–108. IEEE (2014)
66. Chen, S., Langner, C.A., Mendoza-Denton, R.: When dispositional and role power fit: implications for self-expression and self-other congruence. Journal of Personality and Social Psychology **96**(3), 710–27 (2009)
67. Chen, T., Bahsoon, R.: Self-adaptive and Sensitivity-aware QoS Modeling for the Cloud. In: Proceedings of the 8th International Symposium on Software Engineering for Adaptive and Self-Managing Systems (SEAMS), pp. 43–52. IEEE (2013). URL http://dl.acm.org/citation.cfm?id=2487336.2487346
68. Chen, T., Bahsoon, R.: Symbiotic and Sensitivity-aware Architecture for Globally-optimal Benefit in Self-adaptive Cloud. In: Proceedings of the 9th International Symposium on Software Engineering for Adaptive and Self-Managing Systems (SEAMS), pp. 85–94. ACM (2014). DOI 10.1145/2593929.2593931. URL http://doi.acm.org/10.1145/2593929.2593931

69. Chen, T., Bahsoon, R., Yao, X.: Online QoS Modeling in the Cloud: A Hybrid and Adaptive Multi-learners Approach. In: 2014 IEEE/ACM 7th International Conference on Utility and Cloud Computing (UCC), pp. 327–336 (2014)
70. Chen, T., Faniyi, F., Bahsoon, R., Lewis, P.R., Yao, X., Minku, L.L., Esterle, L.: The handbook of engineering self-aware and self-expressive systems. Tech. rep., EPiCS EU FP7 project consortium (2014). URL http://arxiv.org/abs/1409.1793. Available via EPiCS website and arXiv
71. Chen, X., Li, X., Wu, H., Qiu, T.: Real-time Object Tracking via CamShift-based Robust Framework. In: Proceedings of the International Conference on Information Science and Technology (ICIST). IEEE (2012)
72. Chow, G.C.T., Grigoras, P., Burovskiy, P., Luk, W.: An efficient sparse conjugate gradient solver using a Beneš permutation network. In: Proceedings of the 24th International Conference on Field Programmable Logic and Applications, pp. 1–7 (2014)
73. Chow, G.C.T., Tse, A.H.T., Jin, Q., Luk, W., Leong, P.H.W., Thomas, D.B.: A mixed precision Monte Carlo methodology for reconfigurable accelerator systems. In: Proceedings of the ACM/SIGDA 20th International Symposium on Field Programmable Gate Arrays, FPGA 2012, Monterey, California, USA, February 22-24, 2012, pp. 57–66 (2012)
74. Christensen, A.L., O'Grady, R., Dorigo, M.: From fireflies to fault-tolerant swarms of robots. IEEE Transactions on Evolutionary Computation **13**(4), 754–766 (2009)
75. Christensen, E., Curbera, F., Meredith, G., Weerawarana, S.: Web Services Description Language (WSDL) 1.1. World Wide Web Consortium (2001)
76. Chu, F., Zaniolo, C.: Fast and light boosting for adaptive mining of data streams. In: Proceedings of the Eighth Pacific-Asia Knowledge Discovery and Data Mining Conference (PAKDD), pp. 282–292. Sydney (2004)
77. Cichowski, A., Madden, C., Detmold, H., Dick, A., Van den Hengel, A., Hill, R.: Tracking Hand-off in Large Surveillance Networks. In: Proceedings of the International Conference Image and Vision Computing, pp. 276–281. IEEE Computer Society Press (2009). DOI 10.1109/IVCNZ.2009.5378396
78. Claus, C., Boutilier, C.: The Dynamics of Reinforcement Learning in Cooperative Multiagent Systems. In: Proceedings of the Conference on Artificial Intelligence/Innovative Applications of Artificial Intelligence, pp. 746–752. American Association for Artificial Intelligence (1998)
79. Collins, N.: The analysis of generative music programs. Organised Sound **13**, 237–248 (2008)
80. Collins, R.T., Liu, Y., Leordeanu, M.: Online selection of discriminative tracking features. IEEE Transactions on Pattern Analysis and Machine Intelligence **27**(10), 1631–1643 (2005). DOI 10.1109/tpami.2005.205
81. Colorni, A., Dorigo, M., Maniezzo, V., et al.: Distributed optimization by ant colonies. In: Proceedings of the first European conference on artificial life, vol. 142, pp. 134–142. Elsevier (1991)
82. Comaniciu, D., Ramesh, V., Meer, P.: Kernel-based object tracking. IEEE Transactions on Pattern Analysis and Machine Intelligence **25**(5) (2003). DOI 10.1109/tpami.2003.1195991
83. Connors, K.: Chemical kinetics: the study of reaction rates in solution. VCH Publishers (1990)
84. Cox, M.: Metacognition in computation: A selected research review. Artificial Intelligence **169**(2), 104–141 (2005)
85. Cramer, T., Schmidl, D., Klemm, M., an Mey, D.: OpenMP Programming on Intel Xeon Phi Coprocessors: An Early Performance Comparison. In: Proceedings of the Many-core Applications Research Community (MARC) Symposium, pp. 38–44. Aachen, Germany (2012)
86. Crockford, D.: The application/json Media Type for JavaScript Object Notation (JSON). RFC 7159, RFC Editor (2014). URL http://tools.ietf.org/pdf/rfc7159.pdf
87. Curreri, J., Stitt, G., George, A.D.: High-level synthesis of in-circuit assertions for verification, debugging, and timing analysis. International Journal of Reconfigurable Computing **2011**, 1–17 (2011). DOI http://dx.doi.org/10.1155/2011/406857

88. Czajkowski, T.S., Aydonat, U., Denisenko, D., Freeman, J., Kinsner, M., Neto, D., Wong, J., Yiannacouras, P., Singh, D.P.: From OpenCL to high-performance hardware on FPGAs. In: Proceedings of the 22nd International Conference on Field Programmable Logic and Applications (FPL), pp. 531–534. Oslo, Norway (2012)
89. Datta, K., Murphy, M., Volkov, V., Williams, S., Carter, J., Oliker, L., Patterson, D., Shalf, J., Yelick, K.: Stencil computation optimization and auto-tuning on state-of-the-art multicore architectures. In: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis (SC 2008)., p. 4. IEEE (2008)
90. Davidson, A.A., Owens, J.D.: Toward techniques for auto-tuning GPU algorithms. In: Proceedings of the 10th International Conference on Applied Parallel and Scientific Computing (PARA), Revised Selected Papers, Part II, pp. 110–119. Reykjavík (2010)
91. Day, J.: Patterns in Network Architecture: A Return to Fundamentals. Prentice Hall International (2008)
92. Day, J., Matta, I., Mattar, K.: Networking is IPC: A Guiding Principle to a Better Internet. In: Proceedings of the 2008 ACM CoNEXT Conference, pp. 67:1–67:6 (2008). DOI 10.1145/1544012.1544079. URL http://doi.acm.org/10.1145/1544012.1544079
93. Dean, J., Ghemawat, S.: MapReduce: Simplified Data Processing on Large Clusters. In: Proceedings of the 6th Symposium on Operating System Design and Implementation (OSDI), pp. 137–150. San Francisco, California, USA (2004)
94. Deb, K.: Multi-objective optimization using evolutionary algorithms, vol. 16. John Wiley & Sons, England (2001)
95. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Transactions on Evolutionary Computation **6**(2), 182–197 (2002)
96. Denholm, S., Inoue, H., Takenaka, T., Luk, W.: Application-specific customisation of market data feed arbitration. In: Proceedings of the International Conference on Field Programmable Technology (ICFPT), pp. 322–325. IEEE (2013)
97. Denholm, S., Inouey, H., Takenakay, T., Becker, T., Luk, W.: Low latency FPGA acceleration of market data feed arbitration. In: Proceedings of the International Conference on Application-Specific Systems, Architectures, and Processors (ASAP), pp. 36–40. IEEE (2014). DOI 10.1109/ASAP.2014.6868628
98. Dennett, D.C.: Consciousness Explained. Penguin Science (1993)
99. Dennis, J.B., Misunas, D.: A preliminary architecture for a basic data flow processor. In: Proceedings of the 2nd Annual Symposium on Computer Architecture, pp. 126–132 (1974)
100. Dieber, B., Simonjan, J., Esterle, L., Rinner, B., Nebehay, G., Pflugfelder, R., Fernandez, G.J.: Ella: Middleware for multi-camera surveillance in heterogeneous visual sensor networks. In: Proceedings of the International Conference on Distributed Smart Cameras (ICDSC) (2013). DOI 10.1109/ICDSC.2013.6778223
101. Dietterich, T.G.: Ensemble methods in machine learning. In: Proceedings of the First International Workshop on Multiple Classifier Systems, Lecture Notes in Computer Science, pp. 1–15. Springer-Verlag (2000)
102. Diguet, J.P., Eustache, Y., Gogniat, G.: Closed-loop–based Self-adaptive Hardware/Software-Embedded Systems: Design Methodology and Smart Cam Case Study. ACM Transactions on Embedded Computing Systems **10**(3), 1–28 (2011)
103. Dinh, M.N., Abramson, D., J. Chao, D.K., Gontarek, A., Moench, B., DeRose, L.: Debugging scientific applications with statistical assertions. Procedia Computer Science **9**(0), 1940––1949 (2012)
104. Dobson, S., Denazis, S., Fernández, A., Gaïti, D., Gelenbe, E., Massacci, F., Nixon, P., Saffre, F., Schmidt, N., Zambonelli, F.: A survey of autonomic communications. ACM Transactions on Autonomous and Adaptive Systems **1**(2), 223–259 (2006)
105. Dobson, S., Sterritt, R., Nixon, P., Hinchey, M.: Fulfilling the vision of autonomic computing. IEEE Computer **43**(1), 35 –41 (2010)
106. Dobzhansky, T., Hecht, M., Steere, W.: On some fundamental concepts of evolutionary biology. Evolutionary Biology **2**, 1–34 (1968)
107. Dorigo, M.: Optimization, learning and natural algorithms. Ph.D. thesis, Politecnico di Milano (1992)

108. Dorigo, M., Blum, C.: Ant colony optimization theory: A survey. Theoretical computer science **344**(2), 243–278 (2005)
109. Dorigo, M., Maniezzo, V., Colorni, A.: Ant system: optimization by a colony of cooperating agents. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics **26**(1), 29–41 (1996)
110. Dutta, R., Rouskas, G., Baldine, I., Bragg, A., Stevenson, D.: The SILO Architecture for Services Integration, controL, and Optimization for the Future Internet. In: Proceedings of the IEEE International Conference on Communications (ICC), pp. 1899–1904 (2007). DOI 10.1109/ICC.2007.316
111. Duval, S., Wicklund, R.A.: A theory of objective self awareness. Academic Press (1972)
112. Ehrgott, M.: Other Methods for Pareto Optimality. In: Multicriteria Optimization, Lecture Notes in Economics and Mathematical Systems, vol. 491, pp. 77–102. Springer (2000)
113. Eiben, A.E., Smith, J.E.: Introduction to evolutionary computing. Springer (2003)
114. Eigenfeldt, A., Pasquier, P.: Considering vertical and horizontal context in corpus-based generative electronic dance music. In: Proceedings of the Fourth International Conference on Computational Creativity, p. 72 (2013)
115. Eigenfeldt, A., Pasquier, P.: Evolving structures for electronic dance music. In: Proceedings of the 15th Annual Conference on Genetic and Evolutionary Computation (GECCO), pp. 319–326. ACM (2013)
116. Elkhodary, A., Esfahani, N., Malek, S.: FUSION: a framework for engineering self-tuning self-adaptive software systems. In: Proceedings of the eighteenth ACM SIGSOFT International Symposium on Foundations of Software Engineering, pp. 7–16. ACM (2010). DOI 10.1145/1882291.1882296. URL http://doi.acm.org/10.1145/1882291.1882296
117. Elliott, G.T., Tomlinson, B.: PersonalSoundtrack: context-aware playlists that adapt to user pace. In: Proceedings of CHI'06 Extended Abstracts on Human Factors in Computing Systems, pp. 736–741. ACM (2006)
118. Ellis, T., Makris, D., Black, J.: Learning a Multi-camera Topology. In: Proceedings of the Joint International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance, pp. 165–171. IEEE Computer Society Press (2003)
119. Elwell, R., Polikar, R.: Incremental learning of concept drift in nonstationary environments. IEEE Transactions on Neural Networks **22**, 1517–1531 (2011)
120. Endo, T., Matsuoka, S.: Massive supercomputing coping with heterogeneity of modern accelerators. In: Proceedings of the 22nd IEEE International Symposium on Parallel and Distributed Processing (IPDPS), pp. 1–10 (2008)
121. Erdem, U.M., Sclaroff, S.: Look there! Predicting Where to Look for Motion in an Active Camera Network. In: Proceedings of the IEEE Conference on Advanced Video and Signal-based Surveillance, pp. 105–110. Como, Italy (2005)
122. Esterle, L., Lewis, P.R., Bogdanski, M., Rinner, B., Yao, X.: A socio-economic approach to online vision graph generation and handover in distributed smart camera networks. In: Proceedings of the International Conference on Distributed Smart Cameras (ICDSC), pp. 1–6. IEEE (2011). DOI 10.1109/ICDSC.2011.6042902
123. Esterle, L., Lewis, P.R., Caine, H., Yao, X., Rinner, B.: CamSim: A distributed smart camera network simulator. In: Proceedings of the International Conference on Self-Adaptive and Self-Organizing Systems Workshops, pp. 19–20. IEEE Computer Society Press (2013). DOI 10.1109/SASOW.2013.11
124. Esterle, L., Lewis, P.R., Rinner, B., Yao, X.: Improved adaptivity and robustness in decentralised multi-camera networks. In: Proceedings of the International Conference on Distributed Smart Cameras, pp. 1–6. ACM (2012)
125. Esterle, L., Lewis, P.R., Yao, X., Rinner, B.: Socio-economic vision graph generation and handover in distributed smart camera networks. ACM Transactions on Sensor Networks **10**(2), 20:1–20:24 (2014). DOI 10.1145/2530001
126. Eugster, P.T., Felber, P.A., Guerraoui, R., Kermarrec, A.M.: The Many Faces of Publish/Subscribe. ACM Computing Surveys **35**(2), 114–131 (2003)

127. Faniyi, F., Lewis, P.R., Bahsoon, R., Xao, X.: Architecting self-aware software systems. In: Proceedings of the IEEE/IFIP Conference on Software Architecture (WICSA), pp. 91–94. IEEE (2014)
128. Farrell, R., Davis, L.S.: Decentralized discovery of camera network topology. In: Proceedings of the International Conference on Distributed Smart Cameras, pp. 1–10. IEEE Computer Society Press (2008). DOI 10.1109/ICDSC.2008.4635696
129. Fels, S., Hinton, G.: Glove-talk: A neural network interface between a data-glove and a speech synthesizer. IEEE Transactiona on Neural Networks **4**(1), 2–8 (1993)
130. Feng, W.: Making a case for efficient supercomputing. ACM Queue **1**(7), 54–64 (2003)
131. Fenigstein, A., Scheier, M.F., Buss, A.H.: Public and private self-consciousness: Assessment and theory. Journal of Consulting and Clinical Psychology **43**(4), 522–527 (1975)
132. Fern, A., Givan, R.: Online ensemble learning: An empirical study. Machine Learning **53**(1–2), 71–109 (2003)
133. Fette, B.: Cognitive radio technology. Academic Press (2009)
134. Fiebrink, R., Trueman, D., Cook, P.R.: A meta-instrument for interactive, on-the-fly machine learning. In: Proceedings of the International Conference on New Interfaces for Musical Expression. Pittsburgh (2009)
135. Fielding, R.T., Taylor, R.N.: Principled design of the modern web architecture. ACM Transactions on Internet Technology **2**(2), 115–150 (2002). DOI 10.1145/514183.514185. URL http://doi.acm.org/10.1145/514183.514185
136. Freund, Y., Schapire, R.E.: Experiments with a new boosting algorithm. In: Proceedings of the 13th International Conference on Machine Learning, pp. 148–156 (1996)
137. Froming, W.J., Walker, G.R., Lopyan, K.J.: Public and private self-awareness: When personal attitudes conflict with societal expectations. Journal of Experimental Social Psychology **18**(5), 476 – 487 (1982). DOI 10.1016/0022-1031(82)90067-1
138. Fu, H., Sendhoff, B., Tang, K., Yao, X.: Finding robust solutions to dynamic optimization problems. In: Proceedings of the 16th European conference on Applications of Evolutionary Computation (EvoApplications), pp. 616–625 (2013)
139. Funie, A., Salmon, M., Luk, W.: A hybrid genetic-programming swarm-optimisation approach for examining the nature and stability of high frequency trading strategies. In: Proceedings of the 13th International Conference on Machine Learning and Applications (ICMLA), pp. 29–34. Detroit, USA (2014). DOI 10.1109/ICMLA.2014.11. URL http://dx.doi.org/10.1109/ICMLA.2014.11
140. Gallup, G.G.: Chimpanzees: self-recognition. Science (1970)
141. Gama, J., Medas, P., Castillo, G., Rodrigues, P.: Learning with drift detection. In: Proceedings of the 7th Brazilian Symposium on Artificial Intelligence (SBIA) - Lecture Notes in Computer Science, vol. 3171, pp. 286–295. Springer, São Luiz do Maranhão, Brazil (2004)
142. Gao, J., Fan, W., Han, J.: On appropriate assumptions to mine data streams: Analysis and practice. In: Proceedings of the Seventh IEEE International Conference on Data Mining (ICDM), pp. 143–152 (2007)
143. Garlan, D., Cheng, S.W., Huang, A.C., Schmerl, B., Steenkiste, P.: Rainbow: architecture-based self-adaptation with reusable infrastructure. IEEE Computer **37**(10), 46–54 (2004)
144. Gelenbe, E., Loukas, G.: A self-aware approach to denial of service defence. Computer Networks **51**(5), 1299–1314 (2007)
145. Goto, M.: Active music listening interfaces based on signal processing. In: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, vol. 4, pp. 1441–1444 (2007)
146. Gouin-Vallerand, C., Abdulrazak, B., Giroux, S., Mokhtari, M.: Toward autonomic pervasive computing. In: Proceedings of the 10th International Conference on Information Integration and Web-based Applications & Services, iiWAS '08, pp. 673–676. ACM, New York, NY, USA (2008)
147. Goukens, C., Dewitte, S., Warlop, L.: Me, myself, and my choices: The influence of private self-awareness on preference-behavior consistency. Tech. rep., Katholieke Universiteit Leuven (2007)

148. Grabner, H., Bischof, H.: On-line Boosting and Vision. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 260–267 (2006)
149. Grabner, H., Leistner, C., Bischof, H.: Semi-supervised on-line Boosting for Robust Tracking. In: Proceedings of the European Conference on Computer Vision, Lecture Notes in Computer Science, vol. 5302, pp. 234–247 (2008)
150. Group, K.: The OpenCL specification, version: 1.1. http://www.khronos.org/registry/cl/specs/opencl-1.1.pdf. (Accessed March 8, 2016)
151. Gudger, E.W.: A historical note on the synchronous flashing of fireflies. Science **50**(1286), 188–190 (1919)
152. Gudgin, M., Hadley, M., Mendelsohn, N., Moreau, J.J., Nielsen, H.F., Karmarkar, A., Lafon, Y.: SOAP Version 1.2. World Wide Web Consortium (2007)
153. Guo, C., Luk, W.: Accelerating Maximum Likelihood Estimation for Hawkes Point Processes. In: Proceedings of the International Conference on Field Programmable Logic and Applications (FPL), pp. 1–6. IEEE (2013)
154. Guo, C., Luk, W.: Accelerating parameter estimation for multivariate self-exciting point processes. In: Proceedings of the International Symposium on Field-Programmable Gate Arrays (FPGA), pp. 181–184. ACM (2014). DOI 10.1145/2554688.2554765
155. Haikonen, P.O.: Reflections of consciousness: The mirror test. In: Proceedings of the AAAI Fall Symposium on Consciousness and Artificial Intelligence, pp. 67–71 (2007)
156. Hamid, R., Maddi, S., Johnson, A., Bobick, A., Essa, I., Isbell, C.: A novel sequence representation for unsupervised analysis of human activities. Artificial Intelligence **173**(14), 1221–1244 (2009). DOI DOI: 10.1016/j.artint.2009.05.002
157. Hansen, N.: The CMA evolution strategy: A comparing review. In: J. Lozano, P. Larrañaga, I. Inza, E. Bengoetxea (eds.) Towards a New Evolutionary Computation, *Studies in Fuzziness and Soft Computing*, vol. 192, pp. 75–102. Springer Berlin Heidelberg (2006)
158. Happe, M., Agne, A., Plessl, C.: Measuring and Predicting Temperature Distributions on FPGAs at Run-Time. In: Proceedings of the International Conference on Reconfigurable Computing and FPGAs (ReConFig), pp. 55–60. IEEE Computer Society (2011). DOI 10.1109/ReConFig.2011.59
159. Happe, M., Huang, Y., Keller, A.: Dynamic Protocol Stacks in Smart Camera Networks. In: Proceedings of the International Conference on Reconfigurable Computing and FPGAs (ReConFig), pp. 1–6. IEEE (2014)
160. Happe, M., Traber, A., Keller, A.: Preemptive Hardware Multitasking in ReconOS. In: Proceedings of the International Symposium on Applied Reconfigurable Computing (ARC), Springer (2015)
161. Hart, J.W., Scassellati, B.: Robotic self-modeling. In: J. Pitt (ed.) The Computer After Me, pp. 207–218. Imperial College Press / World Scientific Book (2014)
162. Heath, D., Jarrow, R., Morton, A.: Bond pricing and the term structure of interest rates: A new methodology for contingent claims valuation. Econometrica **60**(1), 77–105 (1992)
163. Hernandez, H., Blum, C.: Distributed graph coloring in wireless ad hoc networks: A lightweight algorithm based on Japanese tree frogs' calling behaviour. In: Proceedings of the 4th Joint IFIP Wireless and Mobile Networking Conference (WMNC), pp. 1–7 (2011)
164. Herzen, B.V.: Signal Processing at 250 MHz Using High-Performance FPGAs. In: Proceedings of the ACM Fifth International Symposium on Field-programmable Gate Arrays, pp. 62–68 (1997)
165. Ho, T.K.: The Random Subspace Method for Constructing Decision Forests. IEEE Transactions on Pattern Analysis and Machine Intelligence **20**(8), 832–844 (1998)
166. Ho, T.K., Hull, J.J., Srihari, S.N.: Decision Combination in Multiple Classifier Systems. IEEE Transactions on Pattern Analysis and Machine Intelligence **16**(1), 66–75 (1994)
167. Ho, T.S.Y., Lee, S.B.: Term Structure Movements and Pricing Interest Rate Contingent Claims. Journal of Finance **41**(5), 1011–1029 (1986)
168. Hoare, C.A.R.: An axiomatic basis for computer programming. Communications of the ACM **12**(10), 576–580 (1969)

169. Hockman, J.A., Wanderley, M.M., Fujinaga, I.: Real-time phase vocoder manipulation by runner's pace. In: Proceedings of the International Conference on New Interfaces for Musical Expression (2009)
170. Hoffmann, H., Eastep, J., Santambrogio, M., Miller, J., Agarwal, A.: Application heartbeats for software performance and health. In: ACM SIGPLAN Notices, vol. 45, pp. 347–348. ACM (2010)
171. Hoffmann, H., Eastep, J., Santambrogio, M.D., Miller, J.E., Agarwal, A.: Application Heartbeats: A Generic Interface for Specifying Program Performance and Goals in Autonomous Computing Environments. In: Proceedings of the International Conference on Autonomic Computing (ICAC) (2010)
172. Hoffmann, H., Holt, J., Kurian, G., Lau, E., Maggio, M., Miller, J.E., Neuman, S.M., Sinangil, M., Sinangil, Y., Agarwal, A., Chandrakasan, A.P., Devadas, S.: Self-aware computing in the Angstrom processor. In: Proceedings of the 49th Annual Design Automation Conference, DAC '12, pp. 259–264. ACM, New York, NY, USA (2012)
173. Hoffmann, H., Maggio, M., Santambrogio, M.D., Leva, A., Agarwal, A.: SEEC: A general and extensible framework for self-aware computing. Tech. Rep. MIT-CSAIL-TR-2011-046, Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology (2011)
174. Holland, B., George, A.D., Lam, H., Smith, M.C.: An analytical model for multilevel performance prediction of Multi-FPGA systems. ACM Transactions on Reconfigurable Technology and Systems **4**(3), 27–28 (2011)
175. Holland, O., Goodman, R.B.: Robots with internal models: A route to machine consciousness? Journal of Consciousness Studies **10**(4), 77–109 (2003)
176. Holopainen, R.: Self-organised sound with autonomous instruments: Aesthetics and experiments. Ph.D. thesis, University of Oslo (2012)
177. Hölzl, M., Wirsing, M.: Towards a system model for ensembles. In: Formal Modeling: Actors, Open Systems, Biological Systems, pp. 241–261. Springer (2011)
178. Hölzl, M., Wirsing, M.: Issues in engineering self-aware and self-expressive ensembles. In: J. Pitt (ed.) The Computer After Me, pp. 37–54. Imperial College Press/World Scientific Book (2014)
179. Horn, J., Nafpliotis, N., Goldberg, D.E.: A niched Pareto genetic algorithm for multiobjective optimization. In: Proceedings of the 1st IEEE Conference on Evolutionary Computation, IEEE World Congress on Computational Intelligence, pp. 82–87 (1994)
180. Horn, P.: Autonomic computing: IBM's perspective on the state of information technology. Armonk, NY, USA. International Business Machines Corporation. (2001)
181. Hosseini, M.J., Ahmadi, Z., Beigy, H.: Using a classifier pool in accuracy based tracking of recurring concepts in data stream classification. Evolving Systems **4**(1), 43–60 (2013)
182. Hsu, C.H., Feng, W.C.: Reducing overheating-induced failures via performance-aware CPU power management. In: Proceedings of the 6th International Conference on Linux Clusters: The HPC Revolution (2005)
183. Hu, F., Evans, J.J.: Power and environment aware control of Beowulf clusters. Cluster Computing **12**, 299–308 (2009)
184. Hu, W., Tan, T., Wang, L., Maybank, S.: A Survey on Visual Surveillance of Object Motion and Behaviors. IEEE Transactions on Systems, Man and Cybernetics, Part C **34**(3), 334–352 (2004)
185. Huang, T., Russell, S.: Object Identification in a Bayesian Context. In: Proceedings of the International Joint Conference on Artificial Intelligence, pp. 1276–1283 (1997)
186. Huebscher, M., McCann, J.: Simulation Model for Self-Adaptive Applications in Pervasive Computing. In: Proceedings of the 15th International Workshop on Database and Expert Systems Applications, pp. 694–698. IEEE Computer Society (2004)
187. Hume, D.: A Treatise of Human Nature. Gutenberg eBook (1739). URL http://www.gutenberg.org/ebooks/4705. (Accessed March 8, 2016)
188. Hunkeler, U., Truong, H.L., Stanford-Clark, A.: MQTT-S–A publish/subscribe protocol for Wireless Sensor Networks. In: Proceedings of the Third International Conference on Communication Systems Software and Middleware and Workshops (COMSWARE), pp. 791–798. IEEE (2008)

189. Hunt, A., Wanderley, M.M., Paradis, M.: The importance of parameter mapping in electronic instrument design. In: Proceedings of the International Conference on New Interfaces for Musical Expression, pp. 1–6. National University of Singapore (2002)
190. IBM: An architectural blueprint for autonomic computing (2003). URL http://www-03.ibm.com/autonomic/pdfs/AC Blueprint White Paper V7.pdf. (Accessed March 8, 2016)
191. Iglesia, D.: MobMuPlat (iOS application). Iglesia Intermedia (2013)
192. Intel: Sophisticated library for vector parallelism. http://software.intel.com/en-us/articles/intel-array-building-blocks/. (Accessed March 8, 2016)
193. Investigating RINA as an Alternative to TCP/IP. URL http://irati.eu. (Accessed March 8, 2016)
194. Ishibuchi, H., Murata, T.: A multiobjective genetic local search algorithm and its application to flowshop scheduling. IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews **28**(3), 392–403 (1998)
195. Ishibuchi, H., Tsukamoto, N., Nojima, Y.: Iterative approach to indicator-based multiobjective optimization. In: Proceedings of the IEEE Congress on Evolutionary Computation, pp. 3967–3974 (2007)
196. Ishibuchi, H., Tsukamoto, N., Nojima, Y.: Evolutionary many-objective optimization. In: Proceedings of the 3rd International Workshop on Genetic and Evolving Systems (GEFS), pp. 47–52. IEEE (2008)
197. James, W.: The principles of psychology. Henry Holt & Co. (1890)
198. Janusevskis, J., Riche, R.L., Ginsbourger, D., Girdziusas, R.: Expected Improvements for the Asynchronous Parallel Global Optimization of Expensive Functions: Potentials and Challenges. In: Y. Hamadi, M. Schoenauer (eds.) Learning and Intelligent Optimization, pp. 413–418. Springer (2012)
199. Javed, O., Khan, S., Rasheed, Z., Shah, M.: Camera Handoff: Tracking in Multiple Uncalibrated Stationary Cameras. In: Proceedings of the Workshop on Human Motion, pp. 113–118. IEEE Computer Society Press (2000). DOI 10.1109/HUMO.2000.897380
200. Javed, O., Rasheed, Z., Shafique, K., Shah, M.: Tracking across Multiple Cameras Disjoint Views. In: Proceedings of IEEE International Conference on Computer Vision, p. 952–957 (2003)
201. Jia, J., Veeravalli, B., Ghose, D.: Adaptive load distribution strategies for divisible load processing on resource unaware multilevel tree networks. IEEE Transactions on Computers **56**(7), 999–1005 (2007)
202. Jin, Q., Becker, T., Luk, W., Thomas, D.: Optimising explicit finite difference option pricing for dynamic constant reconfiguration. In: Proceedings of the International Conference on Field Programmable Logic and Applications (FPL), pp. 165–172 (2012)
203. Jin, Y., Olhofer, M., Sendhoff, B.: A framework for evolutionary optimization with approximate fitness functions. IEEE Transactions on Evolutionary Computation **6**(5), 481–494 (2002)
204. Jones, D.R., Schonlau, M., Welch, W.J.: Efficient global optimization of expensive black-box functions. Journal of Global Optimization **13**(4), 455–492 (1998)
205. Jones, P., Cho, Y., Lockwood, J.: Dynamically optimizing FPGA applications by monitoring temperature and workloads. In: Proceedings of the International Conference on VLSI Design (VLSID). IEEE (2007)
206. Kalal, Z., Mikolajczyk, K., Matas, J.: Tracking-Learning-Detection. IEEE Transactions on Pattern Analysis and Machine Intelligence **34**(7), 1409–1422 (2012)
207. Kalman, R.E.: A New Approach to Linear Filtering and Prediction Problems. Journal of Fluids Engineering **82**(1), 35–45 (1960)
208. Kamil, S., Chan, C., Oliker, L., Shalf, J., Williams, S.: An auto-tuning framework for parallel multicore stencil computations. In: Proceedings of the IEEE International Symposium on Parallel & Distributed Processing (IPDPS), pp. 1–12 (2010)
209. Kang, J., Cohen, I., Medioni, G.: Continuous Tracking within and across Camera Streams. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp. 267–272 (2003)

210. Kant, I.: The critique of pure reason. Gutenberg eBook (1781). URL http://www.gutenberg.org/ebooks/4280. Digital edition 2003 (Accessed March 8, 2016)
211. Kela, J., Korpipää, P., Mäntyjärvi, J., Kallio, S., Savino, G., Jozzo, L., Marca, D.: Accelerometer-based gesture control for a design environment. Personal and Ubiquitous Computing **10**(5), 285–299 (2006)
212. Keller, A., Borkmann, D., Neuhaus, S., Happe, M.: Self-Awareness in Computer Networks. International Journal of Reconfigurable Computing pp. 1–10 (2014). DOI 10.1155/2014/692076
213. Kephart, J.O., Chess, D.M.: The Vision of Autonomic Computing. IEEE Computer **36**(1), 41–50 (2003)
214. Kettnaker, V., Zabith, R.: Bayesian Multi-Camera Surveillance. In: Proceedings of the International Conference on Computer Vision and Pattern Recognition, pp. 117–123 (1999)
215. Khan, M.I., Rinner, B.: Energy-aware task scheduling in wireless sensor networks based on cooperative reinforcement learning. In: Proceedings of the International Conference on Communications Workshops (ICCW). IEEE (2014). DOI 10.1109/ICCW.2014.6881310
216. Khare, V., Yao, X., Deb, K.: Performance scaling of multi-objective evolutionary algorithms. In: Evolutionary Multi-Criterion Optimization, Lecture Notes in Computer Science, vol. 2632, pp. 376–390. Springer (2003)
217. Kim, H.S., Sherman, D.K.: Express yourself: Culture and the effect of self-expression on choice. Journal of Personality and Social Psychology **92**(1), 1–11 (2007). DOI 10.1037/0022-3514.92.1.1
218. Kim, J., Seo, S., Lee, J., Nah, J., Jo, G., Lee, J.: OpenCL as a unified programming model for heterogeneous CPU/GPU clusters. In: Proceedings of the 17th ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming (PPOPP), pp. 299–300 (2012)
219. Kittler, J., Hatef, M., Duin, R.P., Matas, J.: On combining classifiers. IEEE Transactions on Pattern Analysis and Machine Intelligence **20**(3), 226–239 (1998)
220. Klinglmayr, J., Bettstetter, C.: Self-organizing synchronization with inhibitory-coupled oscillators: Convergence and robustness. ACM Transactions on Autonomous and Adaptive Systems **7**(3), 30:1–30:22 (2012)
221. Klinglmayr, J., Kirst, C., Bettstetter, C., Timme, M.: Guaranteeing global synchronization in networks with stochastic interactions. New Journal of Physics **14**(7), 1–13 (2012)
222. Knutzen, H., Nymoen, K., Torresen, J.: PheroMusic [iOS application]. URL http://itunes.apple.com/app/pheromusic/id910100415
223. Kolter, J.Z., Maloof, M.A.: Dynamic weighted majority: An ensemble method for drifting concepts. Journal of Machine Learning Research **8**, 2755–2790 (2007)
224. Koski, J., Silvennoinen, R.: Norm methods and partial weighting in multicriterion optimization of structures. International Journal for Numerical Methods in Engineering **24**(6), 1101–1121 (1987)
225. Kramer, J., Magee, J.: Self-managed systems: an architectural challenge. In: Future of Software Engineering (FoSE), pp. 259–268. IEEE (2007)
226. Krishnamoorthy, S., Baskaran, M., Bondhugula, U., Ramanujam, J., Rountev, A., Sadayappan, P.: Effective automatic parallelization of stencil computations. In: Proceedings of the 28th ACM SIGPLAN Conference on Programming Language Design and Implementation, pp. 235–244 (2007)
227. Kuhn, H.W., Yaw, B.: The Hungarian Method for the Assignment Problem. Naval Research Logistics Quarterly pp. 83–97 (1955)
228. Kuncheva, L.I.: A theoretical study on six classifier fusion strategies. IEEE Transactions on Pattern Analysis and Machine Intelligence **24**(2), 281–286 (2002)
229. Kurek, M., Becker, T., Chau, T.C., Luk, W.: Automating Optimization of Reconfigurable Designs. In: Proceedings of the International Symposium on Field-Programmable Custom Computing Machines (FCCM), pp. 210–213. IEEE (2014). DOI 10.1109/FCCM.2014.65
230. Kurek, M., Becker, T., Luk, W.: Parametric Optimization of Reconfigurable Designs Using Machine Learning. In: Proceedings of the International Conference on Reconfigurable Computing: Architectures, Tools and Applications (ARC), *Lecture Notes in Computer Science*, vol. 7806, pp. 134–145. Springer (2013)

231. Legrain, L., Cleeremans, A., Destrebecqz, A.: Distinguishing three levels in explicit self-awareness. Consciousness and Cognition **20**, 578–585 (2011)
232. Legrand, D.: Pre-reflective self-as-subject from experiential and empirical perspectives. Consciousness and Cognition **16**(3), 583–599 (2007)
233. Leidenfrost, R., Elmenreich, W.: Firefly clock synchronization in an 802.15. 4 wireless network. EURASIP Journal on Embedded Systems **2009**, 7:1–7:17 (2009)
234. Leland, W., Taqqu, M., Willinger, W., Wilson, D.: On the self-similar nature of ethernet traffic (extended version). IEEE/ACM Transactions on Networking **2**(1), 1–15 (1994). DOI 10.1109/90.282603
235. Leutenegger, S., Chli, M., Siegwart, R.Y.: BRISK: Binary robust invariant scalable keypoints. In: Proceedings of the International Conference on Computer Vision, pp. 2548–2555. IEEE (2011). DOI 10.1109/iccv.2011.6126542
236. Lewis, P.R., Chandra, A., Faniyi, F., Glette, K., Chen, T., Bahsoon, R., Torresen, J., Yao, X.: Architectural aspects of self-aware and self-expressive computing systems: From psychology to engineering. IEEE Computer **48**(8), 62–70 (2015)
237. Lewis, P.R., Chandra, A., Parsons, S., Robinson, E., Glette, K., Bahsoon, R., Torresen, J., Yao, X.: A Survey of Self-Awareness and Its Application in Computing Systems. In: Proceedings of the International Conference on Self-Adaptive and Self-Organizing Systems Workshops (SASOW), pp. 102–107. IEEE Computer Society, Ann Arbor, MI, USA (2011)
238. Lewis, P.R., Esterle, L., Chandra, A., Rinner, B., Torresen, J., Yao, X.: Static, Dynamic, and Adaptive Heterogeneity in Distributed Smart Camera Networks. ACM Transactions on Autonomous and Adaptive Systems **10**(2), 8:1–8:30 (2015). DOI 10.1145/2764460
239. Lewis, P.R., Esterle, L., Chandra, A., Rinner, B., Yao, X.: Learning to be Different: Heterogeneity and Efficiency in Distributed Smart Camera Networks. In: Proceedings of the International Conference on Self-Adaptive and Self-Organizing Systems (SASO), pp. 209–218. IEEE Computer Society Press (2013). DOI 10.1109/SASO.2013.20
240. Lewis, P.R., Marrow, P., Yao, X.: Resource Allocation in Decentralised Computational Systems: An Evolutionary Market Based Approach. Autonomous Agents and Multi-Agent Systems **21**(2), 143–171 (2010)
241. Lewis, P.R., Platzner, M., Yao, X.: An outlook for self-awareness in computing systems. Awareness Magazine (2012). DOI 10.2417/3201203.004093
242. Li, B., Li, J., Tang, K., Yao, X.: An improved Two Archive Algorithm for Many-Objective Optimization. In: Proceedings of the IEEE Congress on Evolutionary Computation (CEC), pp. 2869–2876 (2014)
243. Li, G., Gopalakrishnan, G.: Scaleable SMT-based verification of GPU kernel functions. In: Proceedings of the Eighteenth International Symposium on the Foundations of Software Engineering (FSE-18) (2010)
244. Li, H., Zhang, Q.: Multiobjective optimization problems with complicated Pareto sets, MOEA/D and NSGA-II. IEEE Transactions on Evolutionary Computation **13**(2), 284–302 (2009)
245. Li, Y., Bhanu, B.: Utility-based Camera Assignment in a Video Network: A Game Theoretic Framework. Sensors Journal **11**(3), 676–687 (2011)
246. Liang, C.J.M., Liu, J., Luo, L., Terzis, A., Zhao, F.: RACNet: A High-Fidelity Data Center Sensing Network. Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems pp. 15–28 (2009)
247. Liu, J., Zhong, L., Wickramasuriya, J., Vasudevan, V.: uWave: Accelerometer-based personalized gesture recognition and its applications. Pervasive and Mobile Computing **5**(6), 657–675 (2009)
248. Liu, Y., Yao, X.: Ensemble learning via negative correlation. Neural Networks **12**(10), 1399–1404 (1999)
249. Lübbers, E., Platzner, M.: Cooperative multithreading in dynamically reconfigurable systems. In: Proceedings of the International Conference on Field Programmable Logic and Applications (FPL), pp. 1–4. IEEE (2009)
250. Lübbers, E., Platzner, M.: ReconOS: Multithreaded programming for reconfigurable computers. ACM Transactions on Embedded Computing Systems **9** (2009)

251. Lucas, B.D., Kanade, T.: An iterative image registration technique with an application to stereo vision. In: Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI), pp. 674–679 (1981)
252. Maggio, M., Hoffmann, H., Santambrogio, M.D., Agarwal, A., Leva, A.: A comparison of autonomic decision making techniques. Tech. Rep. MIT-CSAIL-TR-2011-019, Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology (2011)
253. Makris, D., Ellis, T., Black, J.: Bridging the Gaps between Cameras. In: Proceedings of Conference on Computer Vision and Pattern Recognition, vol. 2 (2004)
254. Marler, R.T., Arora, J.S.: Function-transformation methods for multi-objective optimization. Engineering Optimization **37**(6), 551–570 (2005)
255. Marrow, P.: Nature-inspired computing technology and applications. BT Technology Journal **18**(4), 13–23 (2000)
256. Marsaglia, G., Bray, T.A.: A convenient method for generating normal variables. SIAM Review **6**(3), 260–264 (1964)
257. Masahiro, N., Takaesu, H., Demachi, H., Oono, M., Saito, H.: Development of an automatic music selection system based on runner's step frequency. In: Proceedings of the 2008 International Conference on Music Information Retrieval, pp. 193–8 (2008)
258. Massie, M.L., Chun, B.N., Culler, D.E.: The Ganglia distributed monitoring system: design, implementation, and experience. Parallel Computing **30**, 817–840 (2004)
259. Mathar, R., Mattfeldt, J.: Pulse-coupled decentral synchronization. SIAM Journal on Applied Mathematics **56**(4), 1094–1106 (1996)
260. Max [computer software]. URL http://cycling74.com. (Accessed March 8, 2016)
261. McKay, M.D., Beckman, R.J., Conover, W.J.: A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. Technometrics pp. 55–61 (2000)
262. Mehta, N.R., Medvidovic, N.: Composing architectural styles from architectural primitives. In: Proceedings of the European Software Engineering Conference and ACM SIGSOFT Symposium on the Foundations of Software Engineering, pp. 347–350 (2003). URL http://dblp.uni-trier.de/db/conf/sigsoft/fse2003.html#MehtaM03
263. Menasce, D.A., Sousa, J.a.P., Malek, S., Gomaa, H.: QoS Architectural Patterns for Self-architecting Software Systems. In: Proceedings of the 7th International Conference on Autonomic Computing (ICAC), pp. 195–204. ACM (2010). DOI 10.1145/1809049.1809084
264. Metcalfe, J., Shimamura, A.P. (eds.): Metacognition: Knowing about knowing. MIT Press, Cambridge, MA, USA (1994)
265. Michalski, R.S.: A Theory and Methodology of Inductive Learning. In: Machine Learning, Symbolic Computation, pp. 83–134. Springer Berlin Heidelberg (1983)
266. Miettinen, K., Mäkelä, M.M.: Interactive bundle-based method for nondifferentiable multiobjective optimization: nimbus. Optimization Journal **34**(3), 231–246 (1995)
267. Minku, L.L.: Online ensemble learning in the presence of concept drift. Ph.D. thesis, School of Computer Science, University of Birmingham, Birmingham, UK (2010)
268. Minku, L.L., Yao, X.: DDD: A new ensemble approach for dealing with concept drift. IEEE Transactions on Knowledge and Data Engineering **24**(4), 619–633 (2012)
269. Minku, L.L., Yao, X.: Software Effort Estimation as a Multi-objective Learning Problem. ACM Transactions on Software Engineering and Methodology **22**(4), 35:1–32 (2013)
270. Miranda, E.R., Wanderley, M.: New Digital Musical Instruments: Control and Interaction Beyond the Keyboard. A-R Editions, Inc., Middleton, WI (2006)
271. Mirollo, R.E., Strogatz, S.H.: Synchronization of pulse-coupled biological oscillators. SIAM Journal on Applied Mathematics **50**(6), 1645–1662 (1990)
272. Mitchell, M.: Self-awareness and control in decentralized systems. In: Proceedings of the AAAI Spring Symposium on Metacognition in Computation (2005). Available at http://www.cs.pdx.edu/ mm/self-awareness.pdf
273. Modler, P.: Neural networks for mapping hand gestures to sound synthesis parameters, vol. 18, p. 14. IRCAM — Centre Pompidou (2000)

274. Moens, B., van Noorden, L., Leman, M.: D-Jogger: Syncing music with walking. In: Proceedings of the Sound and Music Computing Conference, pp. 451–456. Barcelona, Spain (2010)
275. Morin, A.: Levels of consciousness and self-awareness: A comparison and integration of various neurocognitive views. Consciousness and Cognition **15**(2), 358–71 (2006)
276. Morin, A., Everett, J.: Conscience de soi et langage interieur: Quelques speculations. [Self-awareness and inner speech: Some speculations]. Philosophiques **XVII**(2), 169–188 (1990)
277. Müller-Schloer, C., Schmeck, H., Ungerer, T.: Organic computing: a paradigm shift for complex systems. Springer (2011)
278. Nakashima, H., Aghajan, H., Augusto, J.C.: Handbook of ambient intelligence and smart environments. Springer (2009)
279. Narukawa, K., Tanigaki, Y., Ishibuchi, H.: Evolutionary many-objective optimization using preference on hyperplane. In: Proceedings of the 2014 Conference on Genetic and Evolutionary Computation Companion, pp. 91–92. ACM (2014)
280. Natarajan, P., Atrey, P.K., Kankanhalli, M.: Multi-camera coordination and control in surveillance systems: A survey. ACM Transactions on Multimedia Computing, Communications and Applications **11**(4), 57:1–57:30 (2015). DOI 10.1145/2710128
281. Nebehay, G., Chibamu, W., Lewis, P.R., Chandra, A., Pflugfelder, R., Yao, X.: Can diversity amongst learners improve online object tracking? In: Z.H. Zhou, F. Roli, J. Kittler (eds.) Multiple Classifier Systems, *Lecture Notes in Computer Science*, vol. 7872, pp. 212–223. Springer (2013). DOI 10.1007/978-3-642-38067-9_19
282. Nebehay, G., Pflugfelder, R.: Consensus-based matching and tracking of keypoints for object tracking. In: Proceedings of the Winter Conference on Applications of Computer Vision (WACV). IEEE (2014)
283. Nebro, A.J., Luna, F., Alba, E., Beham, A., Dorronsoro, B.: AbYSS: adapting scatter search for multiobjective optimization. Tech. Rep. ITI-2006-2, Departamento de Lenguajes y Ciencias de la Computación, University of Málaga, Malaga (2006)
284. Neisser, U.: The Roots of Self-Knowledge: Perceiving Self, It, and Thou. Annals of the NY AoS. **818**, 19–33 (1997)
285. netem. URL http://www.linuxfoundation.org/collaborate/workgroups/networking/netem. (Accessed March 8, 2016)
286. Nguyen, A., Satish, N., Chhugani, J., Kim, C., Dubey, P.: 3.5-D blocking optimization for stencil computations on modern CPUs and GPUs. In: Proceedings of the ACM/IEEE International Conference for High Performance Computing, Networking, Storage and Analysis, pp. 1–13 (2010)
287. Niezen, G., Hancke, G.P.: Evaluating and optimising accelerometer-based gesture recognition techniques for mobile devices. In: Proceedings of AFRICON, pp. 1–6. IEEE (2009)
288. Nishida, K.: Learning and detecting concept drift. Ph.D. thesis, Hokkaido University (2008). URL http://lis2.huie.hokudai.ac.jp/ knishida/paper/nishida2008-dissertation.pdf
289. Nishida, K., Yamauchi, K.: Detecting concept drift using statistical testing. In: Proceedings of the Tenth International Conference on Discovery Science (DS) - Lecture Notes in Artificial Intelligence, vol. 3316, pp. 264–269. Sendai, Japan (2007)
290. Niu, X., Chau, T.C.P., Jin, Q., Luk, W., Liu, Q.: Automating elimination of idle functions by run-time reconfiguration. In: Proceedings of the 21st IEEE Annual International Symposium on Field-Programmable Custom Computing Machines (FCCM), pp. 97–104 (2013)
291. Niu, X., Coutinho, J.G.F., Luk, W.: A scalable design approach for stencil computation on reconfigurable clusters. In: Proceedings of the 23rd International Conference on Field programmable Logic and Applications (FPL), pp. 1–4 (2013)
292. Niu, X., Jin, Q., Luk, W., Liu, Q., Pell, O.: Exploiting run-time reconfiguration in stencil computation. In: Proceedings of the 22nd International Conference on Field programmable Logic and Applications (FPL), pp. 173–180 (2012)
293. Niu, X., Tsoi, K.H., Luk, W.: Reconfiguring distributed applications in FPGA accelerated cluster with wireless networking. In: Proceedings of the 21st International Conference on Field Programmable Logic and Applications (FPL), pp. 545–550 (2011)

294. NVIDIA: Cuda zone. http://www.nvidia.com/object/cuda_home_new.html. (Accessed March 8, 2016)
295. Nymoen, K., Chandra, A., Glette, K., Torresen, J.: Decentralized harmonic synchronization in mobile music systems. In: Proceedings of the International Conference on Awareness Science & Technology (iCAST), pp. 1–6 (2014)
296. Nymoen, K., Chandra, A., Glette, K., Torresen, J., Voldsund, A., Jensenius, A.R.: PheroMusic: Navigating a Musical Space for Active Music Experiences. In: Proceedings of the International Computer Music Conference (ICMC) joint with the Sound and Music Computing Conference, pp. 1715–1718 (2014)
297. Nymoen, K., Song, S., Hafting, Y., Torresen, J.: Funky Sole Music: Gait recognition and adaptive mapping. In: Proceedings of the International Conference on New Interfaces for Musical Expression (NIME), pp. 299–302 (2014)
298. Okuma, K., Taleghani, A., de Freitas, N., Little, J., Lowe, D.: A Boosted Particle Filter: Multitarget Detection and Tracking. In: Proceedings of 8th European Conference on Computer Vision, vol. 3021, pp. 28–39 (2004)
299. Olfati-Saber, R.: Distributed Kalman filtering for sensor networks. In: Proceedings of the Conference on Decision and Control, pp. 5492–5498 (2007). DOI 10.1109/CDC.2007.4434303
300. Olsson, R.A., Keen, A.W.: Remote procedure call. The JR Programming Language: Concurrent Programming in an Extended Java pp. 91–105 (2004)
301. Ong, Y.S., Nair, P.B., Keane, A.J.: Evolutionary optimization of computationally expensive problems via surrogate modeling. AIAA Journal **41**(4), 689–696 (2003)
302. Ontañón, S., Plaza, E.: Multiagent Inductive Learning: An Argumentation-based Approach. In: J. Fürnkranz, T. Joachims (eds.) Proceedings of the 27th International Conference on Machine Learning (ICML), pp. 839–846. Omnipress, Haifa, Israel (2010)
303. Oxford: Oxford dictionaries: Adapt. http://www.oxforddictionaries.com/definition/english/adapt. (Accessed March 8, 2016)
304. Oza, N.C.: Online bagging and boosting. In: Proceedings of the IEEE International Conference on Systems, Man and Cybernetics, pp. 2340–2345 (2005)
305. Oza, N.C., Russell, S.: Experimental comparisons of online and batch versions of bagging and boosting. In: Proceedings of the seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 359–364 (2001)
306. Özuysal, M., Calonder, M., Lepetit, V., Fua, P.: Fast Keypoint Recognition Using Random Ferns. IEEE Transactions on Pattern Analysis and Machine Intelligence **32**(3), 448–461 (2010). DOI 10.1109/tpami.2009.23
307. Page, I., Luk, W.: Compiling occam into Field-Programmable Gate Arrays. In: Proceedings of the International Conference on Field programmable Logic and Applications (FPL) (1991)
308. Papakonstantinou, A., Liang, Y., Stratton, J.A., Gururaj, K., Chen, D., Hwu, W.W., Cong, J.: Multilevel Granularity Parallelism Synthesis on FPGAs. In: Proceedings of the 19th Annual International Symposium on Field-Programmable Custom Computing Machines (FCCM), pp. 178–185. IEEE (2011)
309. Parashar, M., Hariri, S.: Autonomic computing: an overview. In: Proceedings of the International Conference on Unconventional Programming Paradigms, pp. 257–269. Springer-Verlag, Berlin (2005)
310. Parsons, S., Bahsoon, R., Lewis, P.R., Yao, X.: Towards a better understanding of self-awareness and self-expression within software systems. Tech. Rep. CSR-11-03, University of Birmingham, School of Computer Science, UK (2011)
311. Paul, C., Bass, L., Kazman, R.: Software Architecture in Practice. MA: Addison-Wesley (1998)
312. Paul, C., Kazman, R., Klein, M.: Evaluating Software Architectures: Methods and Case Studies. Addison-Wesley (2002)
313. Paulson, L.: DARPA creating self-aware computing. IEEE Computer **36**(3), 24 (2003). DOI 10.1109/MC.2003.1185213
314. Peleg, A., Wilkie, S., Weiser, U.C.: Intel MMX for Multimedia PCs. Communications of the ACM **40**(1), 24–38 (1997)

315. Perkowitz, M., Philipose, M., Fishkin, K., Patterson, D.J.: Mining models of human activities from the Web. In: Proceedings of the 13th International Conference on World Wide Web, pp. 573–582 (2004)
316. Perrone, M., Liu, L.K., Lu, L., Magerlein, K., Kim, C., Fedulova, I., Semenikhin, A.: Reducing Data Movement Costs: Scalable Seismic Imaging on Blue Gene. In: Proceedings of the 26th International Parallel & Distributed Processing Symposium (IPDPS), pp. 320–329 (2012)
317. Perrone, M.P., Cooper, L.N.: When networks disagree: Ensemble methods for hybrid neural networks. Neural Networks for Speech and Image Processing, Chapman-Hall, New York pp. 126–142 (1993)
318. Peskin, C.S.: Mathematical aspects of heart physiology. Courant Institute of Mathematical Sciences, New York University New York (1975)
319. Pflugfelder, R., Bischof, H.: People Tracking across Two Distant Self-calibrated Cameras. In: Proceedings of International Conference on Advanced Video and Signal-based Surveillance. IEEE Computer Society Press (2006)
320. Pflugfelder, R., Bischof, H.: Tracking across Non-overlapping Views Via Geometry. In: Proceedings of the International Conference on Pattern Recognition (2008)
321. Phelps, S., McBurney, P., Parsons, S.: Evolutionary mechanism design: A review. Autonomous Agents and Multi-Agent Systems **21**(2), 237–264 (2010)
322. Piciarelli, C., Esterle, L., Khan, A., Rinner, B., Foresti, G.: Dynamic Reconfiguration in Camera Networks: a short survey. IEEE Transactions on Circuits and Systems for Video Technology **PP**(99), 1–13 (2015). DOI 10.1109/TCSVT.2015.2426575. (early access)
323. Pilato, C., Loiacono, D., Tumeo, A., Ferrandi, F., Lanzi, P.L., Sciuto, D.: Speeding-up expensive evaluations in highlevel synthesis using solution modeling and fitness inheritance. In: Y. Tenne, C.K. Goh (eds.) Computational Intelligence in Expensive Optimization Problems, vol. 2, pp. 701–723. Springer (2010)
324. Polikar, R.: Ensemble based systems in decision making. IEEE Circuits and Systems Magazine **6**(3), 21–45 (2006)
325. Polikar, R., Udpa, L., Udpa, S., Honavar, V.: Learn++: An incremental learning algorithm for supervised neural networks. IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews **31**(4), 497–508 (2001)
326. Puckette, M.: Pure Data (PD) (software). URL http://puredata.info. (Accessed March 8, 2016)
327. Pylvänäinen, T.: Accelerometer based gesture recognition using continuous HMMs. In: Pattern Recognition and Image Analysis, pp. 639–646. Springer (2005)
328. Quaritsch, M., Kreuzthaler, M., Rinner, B., Bischof, H., Strobl, B.: Autonomous Multicamera Tracking on Embedded Smart Cameras. EURASIP Journal on Embedded Systems **2007**(1), 35–45 (2007)
329. Rajko, S., Qian, G., Ingalls, T., James, J.: Real-time gesture recognition with minimal training requirements and on-line learning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1–8. IEEE (2007)
330. Ramamurthy, S., Bhatnagar, R.: Tracking recurrent concept drift in streaming data using ensemble classifiers. In: Proceedings of the Sixth International Conference on Machine Learning and Applications (ICMLA), pp. 404–409. Cincinnati, Ohio (2007)
331. Rammer, I., Szpuszta, M.: Advanced .NET Remoting. Springer (2005)
332. Ramos, C., Augusto, J.C., Shapiro, D.: Ambient intelligence - the next step for artificial intelligence. IEEE Intelligent Systems **23**(2), 15–18 (2008). DOI 10.1109/MIS.2008.19
333. Rasmussen, C., Williams, C.: Gaussian Processes for Machine Learning. MIT Press (2006)
334. Reason [computer software]. URL https://www.propellerheads.se. (Accessed March 8, 2016)
335. ReconOS: A programming model and OS for reconfigurable hardware (2013). URL http://www.reconos.de/. (Accessed March 8, 2016)
336. Reisslein, M., Rinner, B., Roy-Chowdhury, A.: Smart camera networks. IEEE Computer **47**(5), 26–28 (2014)
337. Reyes, R., de Sande, F.: Automatic code generation for gpus in llc. The Journal of Supercomputing **58**(3), 349–356 (2011)

338. Richter, U., Mnif, M., Branke, J., Müller-Schloer, C., Schmeck, H.: Towards a generic observer/controller architecture for organic computing. In: C. Hochberger, R. Liskowsky (eds.) INFORMATIK 2006 – Informatik für Menschen, *LNI*, vol. P-93, pp. 112–119. Bonner Köllen Verlag (2006)
339. Rietmann, M., Messmer, P., Nissen-Meyer, T., Peter, D., Basini, P., Komatitsch, D., Schenk, O., Tromp, J., Boschi, L., Giardini, D.: Forward and adjoint simulations of seismic wave propagation on emerging large-scale GPU architectures. In: Proceedings of the Conference on High Performance Computing Networking, Storage and Analysis (SC) (2012)
340. Rinner, B., Esterle, L., Simonjan, J., Nebehay, G., Pflugfelder, R., Fernandez, G., Lewis, P.R.: Self-Aware and Self-Expressive Camera Networks. IEEE Computer **48**(7), 33–40 (2015)
341. Rinner, B., Winkler, T., Schriebl, W., Quaritsch, M., Wolf, W.: The evolution from single to pervasive smart cameras. In: Proceedings of the Second ACM/IEEE International Conference on Distributed Smart Cameras (ICDSC), pp. 1–10 (2008). DOI 10.1109/ICDSC.2008.4635674
342. Rinner, B., Wolf, W.: Introduction to Distributed Smart Cameras. Proceedings of the IEEE **96**(10), 1565–1575 (2008). DOI 10.1109/JPROC.2008.928742
343. RNA: Recursive Network Architecture. URL http://www.isi.edu/rna. (Accessed March 8, 2015)
344. Rochat, P.: Five levels of self-awareness as they unfold in early life. Consciousness and Cognition **12**, 717–731 (2003)
345. Russell, S.J., Norvig, P.: Artificial Intelligence - A Modern Approach, 3 edn. Pearson Education (2010)
346. Saaty, T.L.: The Analytical Hierarchical Process. McGraw-Hill (1980)
347. Sakellari, G.: The cognitive packet network: A survey. The Computer Journal **53** (2010)
348. SanMiguel, J.C., Shoop, K., Cavallaro, A., Micheloni, C., Foresti, G.L.: Self-Reconfigurable Smart Camera Networks. IEEE Computer **47**(5), 67–73 (2014)
349. Santambrogio, M., Hoffmann, H., Eastep, J., Agarwal, A.: Enabling technologies for self-aware adaptive systems. In: 2010 NASA/ESA Conference on Adaptive Hardware and Systems (AHS), pp. 149–156. IEEE (2010)
350. Santner, J., Leistner, C., Saffari, A., Pock, T., Bischof, H.: PROST: Parallel robust online simple tracking. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 723–730 (2010)
351. Schaumeier, J., Jeremy Pitt, J., Cabri, G.: A tripartite analytic framework for characterising awareness and self-awareness in autonomic systems research. In: Proceedings of the Sixth IEEE Conference on Self-Adaptive and Self-Organizing Systems Workshops (SASOW), pp. 157–162 (2012)
352. Schlömer, T., Poppinga, B., Henze, N., Boll, S.: Gesture recognition with a Wii controller. In: Proceedings of the 2nd International Conference on Tangible and Embedded Interaction, pp. 11–14. ACM (2008)
353. Schmeck, H.: Organic computing - a new vision for distributed embedded systems. In: Proceedings of the Eighth IEEE International Symposium on Object-Oriented Real-Time Distributed Computing (ISORC), pp. 201–203 (2005)
354. Schmickl, T., Thenius, R., Moslinger, C., Timmis, J., Tyrrell, A., Read, M., Hilder, J., Halloy, J., Campo, A., Stefanini, C., Manfredi, L., Orofino, S.: CoCoRo–The Self-Aware Underwater Swarm. In: Proceedings of the International Conference on Self-Adaptive and Self-Organizing Systems Workshops (SASOW), pp. 120–126. IEEE Computer Society, Ann Arbor, MI, USA (2011)
355. Schnier, T., Yao, X.: Using negative correlation to evolve fault-tolerant circuits. In: Proceedings of the 5th International Conference on Evolvable Systems: From Biology to Hardware (ICES'2003) – Lecture Notes in Computer Science, vol. 2606, pp. 35–46. Springer-Verlag (2003)
356. Scholz, M., Klinkenberg, R.: Boosting classifiers for drifting concepts. Intelligent Data Analysis **11**(1), 3–28 (2007)
357. Sharan, K.: Java remote method invocation. In: Beginning Java 8 APIs, Extensions and Libraries, chap. 7, pp. 525–548. Springer (2014)

358. Shaw, M.J., Sikora, R.: A distributed problem-solving approach to inductive learning. Tech. Rep. CMU-RI-TR-90-262, School of Computer Science, Carnegie Mellon University (1990)
359. Shipp, C.A., Kuncheva, L.I.: Relationships between combination methods and measures of diversity in combining classifiers. Information Fusion **3**(2), 135–148 (2002)
360. Showerman, M., Enos, J., Pant, A., Kindratenko, V., Steffen, C., Pennington, R., mei Hwu, W.: QP: A heterogeneous multi-acceleator cluster. In: Proceedings of the International Conference on High-Performance Clustered Computing (2009)
361. Shukla, S.K., Yang, Y., Bhuyan, L.N., Brisk, P.: Shared memory heterogeneous computation on PCIe-supported platforms. In: Proceedings of the 23rd International Conference on Field programmable Logic and Applications (FPL), pp. 1–4 (2013)
362. Simonjan, J., Esterle, L., Rinner, B., Nebehay, G., Dominguez, G.F.: Demonstrating autonomous handover in heterogeneous multi-camera systems. In: Proceedings of the International Conference on Distributed Smart Cameras, pp. 43:1–43:3 (2014). DOI 10.1145/2659021.2669474
363. Sironi, F., Bartolini, D.B., Campanoni, S., Cancare, F., Hoffmann, H., Sciuto, D., Santambrogio, M.D.: Metronome: Operating System Level Performance Management via Self-adaptive Computing. In: Proceedings of the Design Automation Conference (DAC). ACM (2012)
364. Sironi, F., Cuoccio, A., Hoffmann, H., Maggio, M., Santambrogio, M.: Evolvable Systems on Reconfigurable Architecture via Self-aware Adaptive Applications. In: Proceedings of the NASA/ESA Conference on Adaptive Hardware and Systems (AHS) (2011). DOI 10.1109/AHS.2011.5963933
365. Sironi, F., Triverio, M., Hoffmann, H., Maggio, M., Santambrogio, M.: Self-aware Adaptation in FPGA-based Systems. In: Proceedings of the International Conference on Field Programmable Logic and Applications. IEEE (2010)
366. Smallwood, J., McSpadden, M., Schooler, J.: The lights are on but no one's home: meta-awareness and the decoupling of attention when the mind wanders. Psychonomic Bulletin and Review **14**(3), 527–533 (2007)
367. Song, S., Chandra, A., Torresen, J.: An ant learning algorithm for gesture recognition with one-instance training. In: Proceedings of the International Congress on Evolutionary Computation (CEC), pp. 2956–2963. IEEE (2013)
368. SRC Computers, L.: SRC-7 MAPstation. Tech. rep., SRC Computers (2009)
369. Srinivas, N., Deb, K.: Multiobjective optimization using nondominated sorting in genetic algorithms. Evolutionary Computation **2**(3), 221–248 (1994)
370. Stanley, K.O.: Learning concept drift with a committee of decision trees. Tech. Rep. UT-AI-TR-03-302, Department of Computer Sciences, University of Texas at Austin (2003)
371. Sterritt, R., Parashar, M., Tianfield, H., Unland, R.: A concise introduction to autonomic computing. Advanced Engineering Informatics **19**(3), 181–187 (2005)
372. Steuer, R.E., Choo, E.U.: An interactive weighted Tchebycheff procedure for multiple objective programming. Mathematical Programming **26**(3), 326–344 (1983)
373. Stone, P.: Layered Learning in Multiagent Systems: A Winning Approach to Robotic Soccer. MIT Press (2000)
374. Strassen, V.: Gaussian elimination is not optimal. Numerische Mathematik pp. 13:354–356 (1969)
375. Street, W., Kim, Y.: A streaming ensemble algorithm (SEA) for large-scale classification. In: Proceedings of the Seventh ACM International Conference on Knowledge Discovery and Data Mining (KDD), pp. 377–382. New York (2001)
376. Strenski, D.: The Cray XD1 computer and its reconfigurable architecture. Tech. rep., Cray Inc. (2005)
377. Strey, A., Bange, M.: Performance Analysis of Intel's MMX and SSE: A Case Study. In: Proceedings of 7th International Euro-Par Conference on Parallel Processing (Euro-Par), pp. 142–147. Manchester, UK (2001)
378. Susanto, K.W., Todman, T., Coutinho, J.G.F., Luk, W.: Design Validation by Symbolic Simulation and Equivalence Checking: A Case Study in Memory Optimization for Image Manipulation, *LNCS*, vol. 5404, pp. 509–520. Springer (2009)

379. Sutter, H.: The free lunch is over: A fundamental turn toward concurrency in software. Dr. Dobb's Journal (2005)
380. Sutton, R.S., Barto, A.G.: Reinforcement Learning: An Introduction. MIT Press (1998)
381. Taj, M., Cavallaro, A.: Distributed and decentralized multi-camera tracking. IEEE Signal Processing Magazine **28**(3), 46–58 (2011)
382. Tawney, G.A.: Feeling and self-awareness. Psyc. Rev. **9**(6), 570 – 596 (1902)
383. Tesauro, G.: Reinforcement learning in autonomic computing: A manifesto and case studies. IEEE Internet Computing **11**(1), 22–30 (2007)
384. Thomas, D., Luk, W.: Non-uniform random number generation through piecewise linear approximations. In: Proceedings of the International Conference on Field Programmable Logic and Applications (FPL), pp. 1–6 (2006)
385. Thomas, D.B., Luk, W.: Credit Risk Modelling using Hardware Accelerated Monte-Carlo Simulation. In: Proceedings of the 16th IEEE International Symposium on Field-Programmable Custom Computing Machines (FCCM), pp. 229–238 (2008)
386. Todman, T., Boehm, P., Luk, W.: Verification of streaming hardware and software codesigns. In: Proceedings of the International Conference on Field Programmable Technology (ICFPT), pp. 147–150. IEEE (2012)
387. Todman, T., Stilkerich, S.C., Luk, W.: Using Statistical Assertions to Guide Self-Adaptive Systems. International Journal of Reconfigurable Computing **2014**, 1–8 (2014). DOI 10.1155/2014/724585
388. Tong, X., Ngai, E.: A ubiquitous publish/subscribe platform for wireless sensor networks with mobile mules. In: Proceedings of the IEEE Eighth International Conference on Distributed Computing in Sensor Systems (DCOSS), pp. 99–108 (2012)
389. Torresen, J., Hafting, Y., Nymoen, K.: A new Wi-Fi based platform for wireless sensor data collection. In: Proceedings of the International Conference on New Interfaces for Musical Expression, pp. 337–340 (2013)
390. Torresen, J., Plessl, C., Yao, X.: Special Issue on "Self-Aware and Self-Expressive Systems". IEEE Computer **48**(7), 45–51 (2015)
391. Touch, J., Pingali, V.: The RNA Metaprotocol. In: Proceedings of the International Conference on Computer Communications and Networks, pp. 1–6 (2008). DOI 10.1109/ICCCN.2008.ECP.46
392. Trucco, E., Plakas, K.: Video Tracking: A Concise Survey. Journal of Oceanic Engineering **31**(2), 520–529 (2006)
393. Tse, A.H.T., Chow, G.C.T., Jin, Q., Thomas, D.B., Luk, W.: Optimising performance of quadrature methods with reduced precision. In: Proceedings of the International Conference on Reconfigurable Computing: Architectures, Tools and Applications (ARC), *Lecture Notes in Computer Science*, vol. 7199, pp. 251–263. Springer (2012). DOI 10.1007/978-3-642-28365-9_21
394. Tse, A.H.T., Thomas, D.B., Tsoi, K.H., Luk, W.: Dynamic scheduling Monte-Carlo framework for multi-accelerator heterogeneous clusters. In: Proceedings of the International Conference on Field-Programmable Technology (FTP), pp. 233–240 (2010)
395. Tsoi, K.H., Luk, W.: Axel: A Heterogeneous Cluster with FPGAs and GPUs. In: Proceedings of the 18th annual ACM/SIGDA International Symposium on Field Programmable Gate Arrays, pp. 115–124 (2010)
396. Tsymbal, A., Pechenizkiy, M., Cunningham, P., Puuronen, S.: Dynamic integration of classifiers for handling concept drift. Information Fusion **9**(1), 56–68 (2008)
397. Vassev, E., Hinchey, M.: Knowledge representation and awareness in autonomic service-component ensembles – state of the art. In: 14th IEEE International Symposium on Object/Component/Service-oriented Real-time Distributed Computing, pp. 110–119 (2011)
398. Vasudevan, S.: What is assertion-based verification? SIGDA E-News **42**(12) (2012)
399. Vermorel, J., Mohri, M.: Multi-Armed Bandit Algorithms and Empirical Evaluation. In: Proceedings of the European Conference on Machine Learning, pp. 437–448. Springer (2005)
400. Vickrey, W.: Counterspeculation, auctions, and competitive sealed tenders. The Journal of Finance **16**(1), 8–37 (1961)

401. Vinoski, S.: CORBA: Integrating diverse applications within distributed heterogeneous environments. IEEE Communications Magazine **35**(2), 46–55 (1997)
402. Volker, L., Martin, D., El Khayaut, I., Werle, C., Zitterbart, M.: A Node Architecture for 1000 Future Networks. In: Proceedings of the IEEE International Conference on Communications (ICC), pp. 1–5 (2009). DOI 10.1109/ICCW.2009.5207996
403. Volker, L., Martin, D., Werle, C., Zitterbart, M., El-Khayat, I.: Selecting Concurrent Network Architectures at Runtime. In: Proceedings of the IEEE International Conference on Communications (ICC), pp. 1–5 (2009). DOI 10.1109/ICC.2009.5199445
404. Wang, J., Brady, D., Baclawski, K., Kokar, M., Lechowicz, L.: The use of ontologies for the self-awareness of the communication nodes. In: Proceedings of the Software Defined Radio Technical Conference (SDR), vol. 3 (2003)
405. Wang, S., Minku, L.L., Yao, X.: A learning framework for online class imbalance learning. In: Proceedings of the IEEE Symposium on Computational Intelligence and Ensemble Learning (CIEL), pp. 36–45 (2013)
406. Wang, S., Minku, L.L., Yao, X.: Online class imbalance learning and its applications in fault detection. International Journal of Computational Intelligence and Applications **12**(1340001), (1–19) (2013)
407. Wang, S., Minku, L.L., Yao, X.: A multi-objective ensemble method for online class imbalance learning. In: Proceedings of the International Joint Conference on Neural Networks (IJCNN), pp. 3311–3318. IEEE (2014). DOI 10.1109/IJCNN.2014.6889545
408. Wang, S., Minku, L.L., Yao, X.: Resampling-based ensemble methods for online class imbalance learning. In: IEEE Transactions on Knowledge and Data Engineering, vol. 27, pp. 1356–1368. IEEE (2015). DOI 10.1109/TKDE.2014.2345380
409. Wang, Z., Tang, K., Yao, X.: A memetic algorithm for multi-level redundancy allocation. IEEE Transactions on Reliability **59**(4), 754–765 (2010)
410. Watson, R.: The Delta-t Transport Protocol: Features and Experience. In: Proceedings 14th Conference on Local Computer Networks, pp. 399–407 (1989). DOI 10.1109/LCN.1989.65288
411. Werner-Allen, G., Tewari, G., Patel, A., Welsh, M., Nagpal, R.: Firefly-inspired sensor network synchronicity with realistic radio effects. In: Proceedings of the 3rd International Conference on Embedded Networked Sensor Systems, pp. 142–153 (2005)
412. Weyns, D., Schmerl, B., Grassi, V., Malek, S., Mirandola, R., Prehofer, C., Wuttke, J., Andersson, J., Giese, H., Gäschka, K.M.: On patterns for decentralized control in self-adaptive systems. In: R. Lemos, H. Giese, H. Müller, M. Shaw (eds.) Software Engineering for Self-Adaptive Systems II, *Lecture Notes in Computer Science*, vol. 7475, pp. 76–107. Springer Berlin Heidelberg (2013)
413. Wikipedia: Adaptation (computer science). http://en.wikipedia.org/wiki/Adaptation. (Accessed March 8, 2016)
414. Winfield, A.: Robots with internal models: a route to self-aware and hence safer robots. In: J. Pitt (ed.) The Computer After Me. Imperial College Press / World Scientific Book (2014)
415. Wolf, W., Ozer, B., Lv, T.: Smart Cameras as Embedded Systems. IEEE Computer **35**(9), 48–53 (2002)
416. Wright, M.: Open Sound Control: an enabling technology for musical networking. Organised Sound **10**(3), 193–200 (2005)
417. Xiao, L., Zhu, Y., Ni, L., Xu, Z.: GridIS: An Incentive-Based Grid Scheduling. In: Proceedings of the 19th IEEE International Parallel and Distributed Processing Symposium, p. 65b (2005). DOI 10.1109/IPDPS.2005.237
418. Xilinx: SDAccel Development Environment. http://www.xilinx.com/products/design-tools/sdx/sdaccel.html. (Accessed March 8, 2016)
419. Ye, J., Dobson, S., McKeever, S.: Situation identification techniques in pervasive computing: A review. Pervasive and Mobile Computing **8**(1) (2012)
420. Yiannacouras, P., Steffan, J.G., Rose, J.: VESPA: portable, scalable, and flexible FPGA-based vector processors. In: Proceedings of the International Conference on Compilers, Architecture, and Synthesis for Embedded Systems, pp. 61–70 (2008)

421. Yilmaz, A., Javed, O., Shah, M.: Object Tracking: A Survey. ACM Computing Surveys **38**(4), 1–45 (2006)
422. Yin, F., D., M., Velastin, S.: Performance evaluation of object tracking algorithms. In: Proceedings of the International Workshop on Performance Evaluation of Tracking and Surveillance (2007)
423. Yin, L., Dong, M., Duan, Y., Deng, W., Zhao, K., Guo, J.: A high-performance training-free approach for hand gesture recognition with accelerometer. Multimedia Tools and Applications pp. 1–22 (2013)
424. Yu, X., Tang, K., Chen, T., Yao, X.: Empirical analysis of evolutionary algorithms with immigrants schemes for dynamic optimization. Memetic Computing **1**(1), 3–24 (2009)
425. Zadeh, L.: Optimality and non-scalar-valued performance criteria. IEEE Transactions on Automatic Control **8**(1), 59–60 (1963)
426. Zagal, J.C., Lipson, H.: Towards self-reflecting machines: Two-minds in one robot. In: Advances in Artificial Life. Darwin Meets von Neumann, *Lecture Notes in Computer Science*, vol. 5777, pp. 156–164. Springer (2011)
427. Zambonelli, F., Bicocchi, N., Cabri, G., Leonardi, L., Puviani, M.: On self-adaptation, self-expression, and self-awareness in autonomic service component ensembles. In: Proceedings of the Fifth IEEE Conference on Self-Adaptive and Self-Organizing Systems Workshops (SASOW), pp. 108 –113 (2011)
428. Zarezadeh, A.A., Bobda, C.: Hardware Middleware for Person Tracking on Embedded Distributed Smart Cameras. Hindawi International Journal of Reconfigurable Computing (2012)
429. Zeppenfeld, J., Bouajila, A., Stechele, W., Bernauer, A., Bringmann, O., Rosenstiel, W., Herkersdorf, A.: Applying ASoC to Multi-core Applications for Workload Management. In: C. Müller-Schloer, H. Schmeck, T. Ungerer (eds.) Organic Computing – A Paradigm Shift for Complex Systems, *Autonomic Systems*, vol. 1, pp. 461–472. Springer Basel (2011)
430. Zhou, A., Qu, B.Y., Li, H., Zhao, S.Z., Suganthan, P.N., Zhang, Q.: Multiobjective evolutionary algorithms: A survey of the state of the art. Swarm and Evolutionary Computation **1**(1), 32–49 (2011)
431. Ziliani, F., Velastin, S., Porikli, F., Marcenaro, L., Kelliher, T., Cavallaro, A., Bruneaut, P.: Performance evaluation of event detection solutions: the CREDS experience. In: Proceedings of the International Conference on Advanced Video and Signal Based Surveillance, pp. 201–206 (2005)
432. Zitzler, E., Deb, K., Thiele, L.: Comparison of Multiobjective Evolutionary Algorithm: Empirical Results. Evolutionary Computation **8**(2), 173–195 (2000)
433. Zitzler, E., Künzli, S.: Indicator-Based Selection in Multiobjective Search. In: Proceedings of the International Conference on Parallel Problem Solving from Nature (PPSN), vol. 3242, pp. 832–842 (2004)
434. Zitzler, E., Laumanns, M., Thiele, L.: SPEA2: Improving the Strength Pareto Evolutionary Algorithm. Tech. Rep. 103, Computer Engineering and Networks Laboratory (TIK), Swiss Federal Institute of Technology (ETH), Zurich (2001)
435. Zitzler, E., Thiele, L.: Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach. IEEE Transactions on Evolutionary Computation **3**(4), 257–271 (1999)
436. Zitzler, E., Thiele, L., Laumanns, M., Fonseca, C.M., da Fonseca, V.G.: Performance assessment of multiobjective optimizers: An analysis and review. IEEE Transactions on Evolutionary Computation **7**(2), 117–132 (2003)