En Plein Air Visual Agents

Marco Gori[1], Marco Lippi[2], Marco Maggini[1], Stefano Melacci[1], and Marcello Pelillo[3]

[1] DIISM, University of Siena {marco,maggini,mela}@diism.unisi.it,
[2] DISI, University of Bologna marco.lippi3@unibo.it,
[3] ECLT / DAIS, University of Venice pelillo@dais.unive.it

Abstract. Nowadays, machine learning is playing a dominant role in most challenging computer vision problems. This paper advocates an extreme evolution of this interplay, where visual agents continuously process videos and interact with humans, just like children, exploiting life–long learning computational schemes. This opens the challenge of *en plein air visual agents*, whose behavior is progressively monitored and evaluated by novel mechanisms, where dynamic man-machine interaction plays a fundamental role. Going beyond classic benchmarks, we argue that appropriate crowd-sourcing schemes are suitable for performance evaluation of visual agents operating in this framework. We provide a proof of concept of this novel view, by showing methods and concrete solutions for en plein air visual agents. Crowdsourcing evaluation is reported, along with a life–long experiment on "The Aristocats" cartoon. We expect that the proposed radically new framework will stimulate related approaches and solutions.

1 Introduction

Nowadays, most computer vision algorithms are designed to successfully tackle specific tasks, such as image classification, object detection and localization, tracking, semantic segmentation, scene parsing [22, 11, 12, 19, 20]. The remarkable scientific results achieved in the last few years have fueled the diffusion of computer vision technologies even in commercial devices such as cameras, tablets, or smartphones.

However, there seems to be a lack of general results when considering the capability of an automatic agent to acquire and successfully exploit vision skills in unrestricted video environments. In particular, the basic task of semantic labeling of pixels in a given video stream has mostly been approached at the frame level, as the outcome of well-established pattern recognition methods working on images. This modality is far from the natural visual interaction experienced by humans with the surrounding environment. The acquisition of visual concepts would have been more difficult if the human cognitive processes had to analyze a stream of shuffled frames: the extraction of symbolic information from images that are not frames of a temporally coherent visual stream would have been extremely harder than in the natural visual experience. Pursuing this idea, we propose studying agents which develop visual skills through a life–long learning process that takes place following a protocol inspired by a human-like communication scheme to deal with unrestricted video. A similar idea is tackled down by the NEIL project [1], but working in the context of images only. In this scenario we propose an in-depth re-thinking of the role of machine learning in computer vision. We argue that a new perspective should be followed facing the challenge of disclosing the computational basis of vision by regarding it as a truly learning field that needs to be attacked by an appropriate *vision learning theory*. In particular, we think that the first step in this direction is to move the target to unrestricted visual environments, and to consider a human-like communication protocol, instead of focusing on brute–force learning on massive labeled datasets of images. We refer to this learning protocol as *learning to see like children* (L2SLC) to stress our view on how visual skills should be acquired. In this framework, there is no neat distinction between learning and test sets, but there is just a visual environment (a video stream) where the agent lives and receives its stimuli.

As pioneer examples of agents implementing the proposed protocol, we describe *Developmental Visual Agents* (DVAs). In these agents, learning is driven by several factors that can be unified under the general concept of *constraint*. The theory of *learning from constraints* [5, 2, 8, 18] allows to incorporate different rich contributions, such as parsimony principles, external supervisions, and complex dependencies among the developed concepts. In particular, we consider motion coherence as a fundamental constraint to reduce the complexity for learning visual skills [10, 7]. In fact, this constraint imposes that any label attached to a moving pixel has to be the same during its motion, thus significantly extending the provided supervisions. This aspect is essentially ignored in most machine learning approaches working on datasets of tagged images. Moreover, DVAs undergo developmental stages, that very much resemble those featured in humans [6], by exploiting a life–long computational scheme[4].

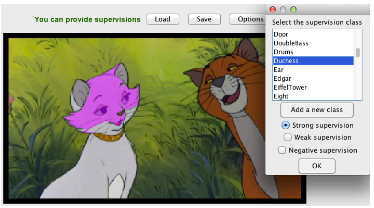
In the L2SLC scenario, the assessment of visual agents by classical benchmark approaches[5] seems unnatural, and, hence, we propose to explore a different experimental validation that seems to resonate perfectly with the considered life–long learning protocol. DVAs are to be tested in unrestricted environments and contexts during their lives. An evaluation scheme based on crowdsourcing is proposed, named *En plein air*. In this open on-line lab, any subscribed user is able to monitor and rate the performance of the currently deployed visual agents. In this paper we show two possible scenarios of assessment: an explicit rating of the performances of different agents in given environments, and the possibility to monitor a set of agents during their learning process. A prototype of our crowdsourcing initiative, our implementation of DVAs, and their outcome in several visual world are collected at `http://dva.diism.unisi.it`.

2 Learning Protocol

We consider visual agents performing scene semantic labeling on a video stream. Given a set $\mathcal{T}$ of semantic categories (*tags*), the agent labels each pixel in a video frame with a subset of $\mathcal{T}$ (i.e. it performs *multi-tag prediction*). Pixel tagging is performed continuously by the visual agent as time flows. The tag set is created by a human supervisor in an incremental way, so that new tags can be added in any moment of the agent's life. The supervisor also provides supervisions to specific patterns in the observed scenes and

[4] The acronym DVA was introduced in [9, 17] in the context of low-level feature extraction and image classification. Here we are extending those agents to the life-long learning processing of video streams, performing semantic labeling.

[5] The risk of biases in vision benchmarks, always recognized, was explicitly pointed out in [23].



You can provide supervisions

Select the supervision class

● Strong supervision

○ Weak supervision

□ Negative supervision

Fig. 1. Interaction of a supervisor with a running visual agent. A semantic label is provided for a region of pixels (highlighted in pink). ("The Aristocats" cartoon, © The Walt Disney Company).

possibly also semantic constraints among the defined tags. Hence, at each time step a tag set $\mathcal{T}_t = \{t_1, \ldots, t_{k_t}\}$ is given, along with a set of constraints $\mathcal{C}_t = \{C_1, \ldots, C_{c_t}\}$ defined on the elements of $\mathcal{T}_t$, possibly depending on the environment configuration (i.e. spatio-temporal variables in the video stream). We do not assume any particular requirement on the nature of constraints, except that they can be specified with a given mathematical formalism. For instance, using First Order Logic we can express ontological constraints among tags, such as the relationship between categories modeled by the *is-a* predicate. Motion coherence can instead constrain the tags assigned to corresponding pixels, given the perceived motion, in a sequence of consecutive frames.

Once the agent is *born*, it starts analyzing the input video stream and to develop its internal model of the experienced environment. Learning begins as soon as the agent is deployed, initially following the constraints imposed by its own architecture and by a set of basic behaviors, such as those deriving from parsimony principles and motion coherence. The learning protocol assumes supervisors to intervene at the symbolic level, by attaching tags to visual patterns in a given frame. Supervisions can be provided at any time step and they are managed asynchronously by the agent as it continues to output predictions on the incoming frames. We consider two different kinds of supervisions to be fed to the agent: (i) *strong*, that specify one or more tags for a specific (group of) pixel(s) in a certain frame, and (ii) *weak*, that express the *presence* of an object, regardless of its specific location in the frame. Figure 1 shows such alternatives in the current version of our interface: the tag *Duchess* can be associated with strong supervision by selecting a region (*here is Duchess*), whereas weak supervision is provided at frame level (*in this frame there is Duchess*). Clearly, weak supervisions require the agent to detect the object to be supervised, and thus they are effective only after enough strong supervisions have been provided, as a reinforcement of visual concepts in their initial stages. Weak supervisions can be useful in real-time scenarios, for example if users provide spoken supervisions through a microphone. Visual agents are also expected to take the initiative by *asking for* supervision, thus exploiting an active learning scheme.

The proposed learning protocol is mainly inspired by the observation that children can learn to recognize objects and actions from a few supervised examples, whereas nowadays machine learning approaches strive to achieve this task without the availability of massive labeled datasets. This difference seems to be deeply rooted in the communication protocol at the basis of the acquisition of visual skills in children and machines. For this reason we refer to the proposed protocol as *learning to see like children*

(L2SLC). The visual agent *lives* in its learning environment (its specific video stream), experiencing a life–long learning process that involves the exploitation of both natural constraints and externally provided teaching signals. Among the natural constraints it seems to be important to include parsimony principles, that constrain the complexity of the solution to be as low as possible, and the need to take coherent decisions with respect to the *perceived motion* of the pixels in the video stream. The linguistic process of attaching symbols to objects takes place at a later stage of children development, when they have already developed strong pattern regularities. We conjecture that, regardless of biology, the enforcement of the motion coherence constraint is a high level computational principle that plays the fundamental role for discovering pattern regularities.

3 *En Plein Air* Assessment

The impressive progress of computer vision has strongly benefited from the massive diffusion of benchmarks which, by and large, are regarded as fundamental tools for performance evaluation. Nowadays, the majority of researchers assume to have access to huge collections of labeled data to evaluate their algorithms, and, when they are not available, they are created from scratch. Despite their apparent indisputable dominant role in the advancements in computer vision, some criticisms have been recently raised [23]. Moreover, this methodology may not always be applicable to the setting of Section 2, at least not in a straightforward way, or it would require overwhelming efforts.

As a matter of fact, the proposed learning protocol involves visual agents operating in dynamic environments. Differently from the case of classical batch data sets, there is no clear distinction between training set and test set. This situation raises a very important question: *which is the right method to evaluate these visual agents?* It is clearly impossible to give a definitive answer, but the previous considerations suggest that the time has come to open the mind towards new approaches. The benchmark–oriented attitude, nowadays dominating the computer vision community, bears some resemblance to the influential testing movement in psychology which has its roots in the turn-of-the-century work of Alfred Binet on IQ tests (see e.g. [21]). Both cases consist in attempts to provide a rigorous way of assessing the performance or the aptitude of a (biological or artificial) system, by agreeing on a set of standardized tests which, from that moment onward, become the ultimate criterion for validity. The IQ testing movement has been severely criticized not only for the social and ethical implications deriving from the idea of ranking human beings on a numerical scale but also, more technically, on the grounds that, irrespective of the care with which such tests are designed, they are inherently unable to capture the multifaceted nature of real-world phenomena. Related concerns were given in the seminal paper by David McClelland [15], that sets the stage for the modern competency movement in the U.S. Motivated by analogous concerns, we maintain that the time is ripe for the computer vision community to adopt a similar grade-in-life attitude towards the evaluation of its systems and algorithms. Clearly, we do not intend to diminish the importance of benchmarks, as they are indeed invaluable tools for the progress of the field. The recently proposed “Visual Turing Tests” [4, 13] share similar ideas, by proposing to assess whether machines can perform scene

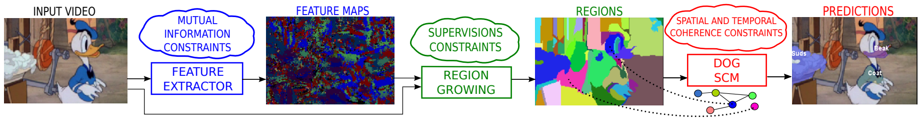


Fig. 2. The pipeline of a Developmental Visual Agent.

understanding as well as humans, for example by answering a list of yes/no questions regarding a given image (i.e., is the boy with the red hat drinking?).

It is clear that the skills of any visual agent can be quickly evaluated and promptly judged by humans, simply by observing its behavior. Thus, we propose a *crowdsourcing performance evaluation scheme* where registered people can inspect and assess the performance of software agents. We use the term *en plein air* ("in the open air"), mimicking the French Impressionist painters of the 19th-century and, more generally, the act of painting outdoors. This term suggests that visual agents should be evaluated by allowing people to see them in action, virtually opening the doors of research labs. A prototype of such evaluation scheme can be experimented at `http://dva.diism.unisi.it`. Registered users can observe the quality of visual agent predictions, and rate them (from 0 up to 5). Scores are then averaged over all the users. Clearly, this kind of mechanism does make sense when the judges have access to the setting in which the agent operates, to better evaluate the difficulty of the task and the impact of the presented results. For this reason, each agent comes with a short description of the experimental setting. This evaluation procedure also allows the users to monitor sets of agents during their life, verifying their progresses in the learning process. This is well-suited for the life–long learning protocol of Section 2, and the first case study will be described in Section 5.

The *en plein air* proposal allows others to test our algorithms and to contribute to this evaluation method by providing their own data, their own results, or the comparisons with their own algorithms. Our web site hosts a software package with a graphical interface which can be used to interact with the agents that we will be describing shortly (Section 4), by providing supervisions and observing the resulting predictions.

4 Pioneer Visual Agents

The learning protocol and the *en plein air* framework allow us to define a general variety of visual agents, designed so as to perform scene understanding in unrestricted domains, following a life–long learning paradigm. Here we introduce a first implementation of pioneer agents within this context, named Developmental Visual Agents (DVAs) [9, 17]. The DVA architecture is hierarchically organized, starting with feature extraction from input visual streams, up to symbolic layers where user interaction occurs. Figure 2 depicts the system pipeline. The learning principles of DVAs are rooted in the theory of *learning from constraints* [5], that allows us to model the interaction of intelligent agents with the environment by means of constraints on the tasks to be learned, and gives foundations and algorithms to discover tasks that are consistent with the given constraints and minimize a parsimony index. The notion of constraint is well-suited to express both visual and linguistic granules of knowledge. Visual constraints can just encode supervisions on a labeled pixel, but the same formalism can represent also motion coherence, or complex dependencies on real-valued functions, including abstract logic formalisms [2]. While this is an ideal view to embrace different visual constraints in the same mathematical and algorithmic framework, we also consider life–long learning computational schemes where the system adapts gradually to the incoming visual stream.

Let $\mathcal{V}$ be a video stream, and $\mathcal{V}_t$ the frame processed at time t. DVA first extracts a stack of L layers of hierarchical scale- and rotation-invariant features, that are developed following the ideas described in [9, 17]. Basically, for each pixel x, the goal is to learn a code of d_ℓ features (for the ℓ-th layer) by fulfilling a constraint driven by information-theoretic principles, that aims at maximizing the mutual information of the code with respect to the observed input, with no interactions with external supervisors. Features are computed over a neighborhood of x at the different levels of the hierarchy, by modeling a *receptive field* of x with a set of $\mathcal{N}$ Gaussians g_k, $k = 1, \ldots, \mathcal{N}$, located nearby the pixel. Thus, higher layers in the hierarchy virtually observe larger input portions. The features of each layer are encoded with probability scores, and the L feature codes are stacked into a single descriptor for pixel x. While in [9, 17] feature extraction injects invariance to geometric transformations by processing image sets, here we follow the strategy of [7] to handle on-line video streams: the data covered by the receptive fields, also referred to as *receptive inputs*, are compared with an internal (geometrically invariant) representation of the video receptive inputs up to time t. This induces a pixel-wise *motion estimation*, a strong basis over which DVAs can learn invariant features.

On top of this unsupervised feature extraction process, DVAs partition the input frame into homogeneous superpixels (regions) to reduce the computational burden of pixel-based tagging. We extend the graph-based region-growing algorithm in [3] by progressively merging pixels according to a dissimilarity score based both on color similarity (as in [3]) and on motion coherence. The dissimilarity is decreased (increased) for those pixels whose estimated motion is (is not) coherent, so that neighbor pixels locally moving in the same direction will more likely belong to the same region. The partitioning obtained for frame $\mathcal{V}_t$ contains a set of R^t regions which correspond to visual patterns that users can tag. Region $r \in R^t$ is described with a histogram z_r, exploiting average feature pooling on the pixels belonging to it. During the agent's life, descriptors are progressively accumulated as vertices (nodes) of a graph, named *Developmental Object Graph* (DOG). We indicate with V_t the set of vertices at time t. To avoid storing duplicate nodes in the DOG, and also to meet practical memory budget requirements, a user-defined tolerance τ between vertices is employed. In detail, after having computed the descriptor z_r, its nearest-neighbor within the current set of DOG vertices is retrieved by the χ^2 distance d_{χ^2}: if $d_{\chi^2} > \tau$ then a new vertex is added to the DOG, otherwise z_r is mapped to (or "hits") its nearest-neighboring vertex. Thus, each region $r \in R^t$ is associated to a node, while the same node can be associated to multiple regions over the video. To meet real-time requirements, we exploit search space partitioning and we allow sub-optimal solutions to speed up nearest-neighbor search.

Two vertices v_i and v_j can be linked by two categories of edges, if one of the following conditions occurs: (i) they are spatially similar; (ii) the agent collected evidence that motion estimation is connecting them. For the first condition, we link nodes whose distance is smaller than a pre-defined threshold γ_s. The *spatial* weight of an edge is computed as $w^s_{ij} = \exp(-\chi^2(v_i, v_j)/2\sigma^2_\tau)$. The second condition involves two consecutive frames $\mathcal{V}_{t-1}$ and $\mathcal{V}_t$. If a region belonging to $\mathcal{V}_{t-1}$ and a similar-sized region of $\mathcal{V}_t$ are such that most of their pixels are connected by the motion estimation procedure, then the *motion-based* weight w^m_{ij} between their associated vertices (v_i and v_j) should be increased. We ignore cases where the number of connected pixels is too small (given a threshold γ_m) with respect to region sizes. During the agent's life, we update w^m_{ij} as long as we accumulate new evidence of motion-based connections between v_i and v_j[6].

The last computational block of Figure 2 involves the symbolic decision mechanism. Labels can be attached by users to the visual patterns stored in the DOG as classic supervisions. For each new semantic tag t_k introduced by a supervisor, a new function $f_k(v_j)$ is created, operating on the space of DOG vertices (we hereby discard the time index, for the sake of simplicity). These functions are defined within the framework of *learning from constraints* [5], which is based on the notion of *constraint*, to model interactions with the environment, and on the parsimony principle. The degree of parsimony of $f = [f_1, \ldots, f_{t_k}]$ is defined by means of a given norm $\|f\|$ [5]. Functions $f_k(v_j)$ have to satisfy coherence constraints defined over the spatio-temporal manifold induced by the DOG structure, as well as supervision constraints. We indicate the penalty associated to supervision constraints with $\mu^{(1)}_{\mathcal{S}}$, and that of coherence constraints as $\mu^{(2)}_{\mathcal{M}}$. Thus, the problem of learning f from (soft) constraints can be formulated as:

$$f^* = \arg\min_f \left\{ \|f\|^2 + \mu^{(1)}_{\mathcal{S}}(f) + \mu^{(2)}_{\mathcal{M}}(f) \right\} \ . \tag{1}$$

It is possible to prove a representer theorem which extends the classical kernel-based representation of traditional learning from examples, leading to the so-called Support Constraint Machine (SCM) [5]. The solution of eq. 1 is then: $f^*_k = \sum_{i=1}^N \zeta_{ik} K(x_i, \cdot)$, being $K(\cdot,\cdot)$ the kernel associated with the selected norm (exponential χ^2 kernel) and ζ_{ik} the parameters to be optimized. Being $\mathcal{S}_k = \{(v_i, y_{i,k}),\ i = 1, \ldots, l_k\}$ the set of supervised DOG nodes for function f_k, and being $y_{i,k} \in \{-1, +1\}$ the label attached to some node $v_i \in V$ for function f_k, the supervision constraint can be expressed as:

$$\mu^{(1)}_{\mathcal{S}}(f) = \sum_{k=1}^{t_k} \sum_{(v_i, y_{i,k}) \in \mathcal{S}_k} \beta_{ik} \max(0, 1 - y_{i,k} f_k(v_i))^2 \ .$$

where t_k is the number of classes for which the agent has received supervisions until time t, and the scalar $\beta_{ik} > 0$ is the *belief* [5] of each point-wise constraint. When a new constraint is added, its belief is set to a fixed initial value. Then, β_{ik} is increased as the same constraint is provided multiple times, while decreased in case of mismatching supervisions, keeping $\sum_i \beta_{ik} = 1$. This mechanism allows the agent to better focus on frequently-provided supervisions, and to give less weight to noisy and incoherent labels. Coherence constraints instead enforce smooth decisions over DOG nodes connected by any kind of edges, leading to an instance of classic manifold regularization [16]:

$$\mu^{(2)}_{\mathcal{M}}(f) = \sum_{k=1}^{t_k} \sum_{i=1}^{|V|} \sum_{j=i+1}^{|V|} w_{ij} (f_k(v_i) - f_k(v_j))^2 \ ,$$

[6] w^m_{ij} is averaged over all the accumulated evidences.

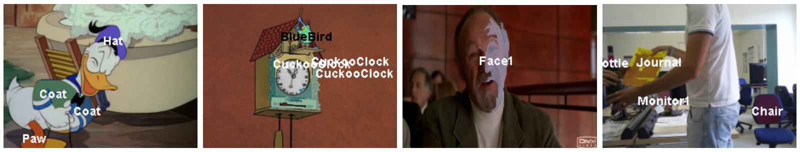


Fig. 3. DVA sample predictions on four visual worlds (left-to-right). Only regions with confidence greater than zero are highlighted (best viewed in colors) and labeled with the most-confident class.

The *belief* of each coherence constraint is a linear combination of edge weights: $w_{ij} = \lambda_{\mathcal{M}} \left(\alpha_{\mathcal{M}} \cdot w^s_{ij} + (1 - \alpha_{\mathcal{M}}) \cdot w^m_{ij} \right)$, where $\lambda_{\mathcal{M}} > 0$ is the global weight of the coherence constraints, and $\alpha_{\mathcal{M}} \in [0, 1]$ balances spatial/motion-based contributions.

As DVAs are expected to react and make predictions at any time, while learning evolves asynchronously, we assume f to operate in a *transductive environment* on the space of DOG nodes[7]. The life–long learning procedure operates by caching values $f_k(v_h)$ over each $v_h \in V$ after each update of ζ_{ik}: in this way agents can continuously make predictions, while the underlying optimization process is still ongoing. To avoid abrupt changes of f, parameters ζ_{ik} associated with newly introduced representatives are set to zero. As memory restrictions are clearly imposed, we must define both a memory budget and a removal policy when the DOG is full: we chose to remove vertices with a small number of hits over a time window. Node hits are also used as frequency indicators for visual patterns, to select vertices upon which *ask* users for supervision.

5 Case Studies

We now describe two different case studies where visual agents that follow the learning protocol of Section 2 are evaluated in the *en plein air* framework described in Section 3. We employ DVAs, but we remark that our proposal can be extended to other instances of visual agents, encouraging other laboratories to promote their own implementations and evaluate them using the principles addressed in this work.

In the first experiment, we aim to compare five DVAs on a set of videos taken from four different visual worlds: a Donald Duck cartoon, a Pink Panther cartoon, a set of (merged) clips from the movie "Get Shorty" (taken from the HoHA2 database [14]) and another real-world video recorded with a fixed webcam in our lab (all of them processed at 240×180 resolution, 25 fps). We chose four heterogenous videos, but DVAs can process any kind of video source. We defined 4–6 semantic classes (from frequently observed objects) for each world[8], and we provided the four DVAs very few supervisions for each class (from 5 up to 10, only positive). Only a first portion (≈ 2 minutes) of each sequence was used to provide supervisions, while the remaining parts (≈ 1 minute) were used to assess the generalization capability of each agent. We asked users to rate each agent on each possible world (independently) with a score in the range

[7] Note that this happens also for feature functions [7].

[8] Donald Duck: {hat, coat, paw, pluto, collar, beak}; Pink Panther: {pink panther, pillow, blanket, blue bird, cuckoo clock}; Get Shorty: {face1, face2, face3, face4}; Webcam: {face1, monitor1, journal, bottle, poster1}. Classes ending with a digit refer to specific instances.

Table 1. Results obtained by the crowdsourcing evaluation process on five DVAs, averaged on all rates (from 0 up to 5) obtained in all four worlds, rescaled as percentages in the last column.

Agent	**Description**	**Rate**	**Rate %**
A1	BA (Base Architecture)	3.16	63.2%
A2	BA without motion constraints, i.e., $\alpha_{\mathcal{M}} = 1$ (see Section 4)	2.55	51.0%
A3	BA with a larger DOG, storing up to 20,000 nodes	2.47	49.4%
A4	BA with double amount of supervisions for each class	3.26	65.2%
A5	BA which processed a ∼2x longer sequence	3.36	67.2%

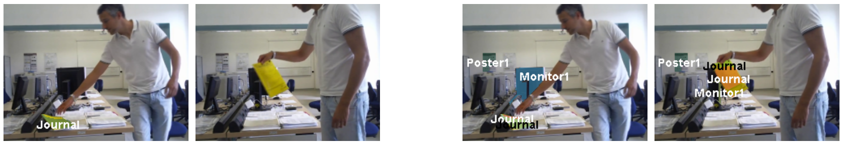


Fig. 4. Sample prediction on two frames without (left) or with (right) motion constraints.

0–5, by evaluating the ability of the system to identify and tag elements in the video stream. Currently, over 40 users, including AI/CV researchers and Computer Science students, were involved in the rating process, but anyone can register on the website and contribute. Figure 3 shows samples of DVA predictions for the four worlds, as shown to the subscribed users: each region was labeled with the most-confident class, highlighting only those regions over which the confidence was greater that zero.

While all the agents share the same settings on the low-level feature extractors[9], they were diversified by high-level characteristics. The first agent (A1) exploits a Base Architecture (BA) designed to store up to 10,000 nodes in the DOG, including spatial and motion-based connections. The settings of the other agents, described in the second column of Table 1, were chosen to evaluate the impact of some specific DVA components: the effect of motion constraints (A2), the use of a larger DOG (A3), a higher number of supervisions (A4), and a longer duration of the agent's life (A5). Table 1 reports the votes collected with this first crowdsourcing evaluation, averaged on all the rates obtained in the four considered worlds. Motion constraint results to be crucial to improve the quality of the agents (A1 vs. A2), as motion constraints can propagate supervisions over DOG nodes connected by motion links. This happens both for moving instances, but also for static objects undergoing small changes in appearance due to illumination or occlusions (see Figure 4). Not surprisingly, also more supervisions (A4) improve the performance, whereas doubling the DOG size (A3) was badly rated. A possible explanation is that a more densely sampled DOG would require appropriate parameter adjustments (e.g., kernel width, spatial/motion constraints weights). Processing longer sequences (A5) also yields better DVAs, because motion links become more stable with time and noisy DOG nodes are filtered out by the long–term removal policy.

[9] The model settings were chosen to fulfill real-time processing on an ordinary multicore CPU: 5×5 receptive fields, 1 layer/feature-category, spatial scales in $\{1, 1.5\}$, 8 in-plane rotation angles, 800 features. See [17, 9] for a detailed description of each parameter.

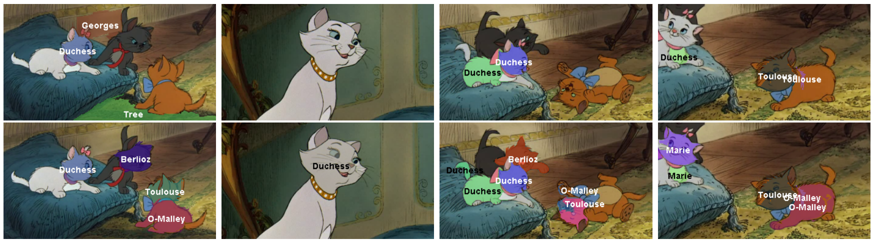


Fig. 5. Sample predictions by DEVA on the same scene at different life stages (top to bottom: 1 day old, 5 days old). Better skills are acquired as long as the agent "grows up".

The second experiment we present is inspired by life–long learning principles and guided by the protocol of Section 2. In this case, the so called agent DEVA was developed, by continuously processing[10] the cartoon "The Aristocats" (© The Walt Disney Company), and by receiving every day new supervisions from a set of selected supervisors. The outcome of DEVA processing can be monitored online for each day in its life (`http://dva.diism.unisi.it/demo_aristocats.html`), to check its evolution, its improvements and common mistakes[11]. A web interface allows to select the day, the semantic classes to visualize, and the agent's sensitivity. Figure 5 shows a result extrapolated from this experiment, where DEVA was tested on the same video sequence at different life stages (1 vs. 5 days old). Despite some errors (Toulouse and O'Malley, both reddish, and Duchess and Marie, both white, are easy to confuse) we can clearly appreciate how DEVA progressively acquires better visual skills during its life. This experiment addresses two distinct issues: (1) we publicly share results on a life–long experiment monitoring the gradual development of an agent; (2) we present a dynamic scenario where the number of classes incrementally grows over time, while existing classes keep receiving supervisions. The experiment lasted about three months.

6 Conclusions

This paper introduced a new perspective in the design and evaluation of agents that acquire visual skills simply by living in their own visual environment and by interacting with humans. A crowd-sourcing based evaluation scheme is proposed that can be instantiated by exploiting different human interaction modalities. *En plein air* visual agents open the doors of research labs all over the world, by allowing any subscribed user to monitor and rate the performance of the currently deployed visual agents. Developmental Visual Agents turn out to be a proof of concept of the general principles outlined in this paper. DEVA, one of these agents, has been watching "The Aristocats" cartoon for months, interacting with humans who provided supervision. Future releases will include the possibility to supervise DEVA by registered users. While this paper reports the first attempt of pioneering the proposed idea, this general scheme is likely to be exploited in other labs by different approaches.

[10] DEVA actually processes a few minutes of video per day, to allow performance analysis.

[11] Processing at 320 × 240, 25fps, 20k DOG nodes, low-levels as in crowdsourcing experiments.

References

1. Xinlei Chen, Abhinav Shrivastava, and Abhinav Gupta. NEIL: Extracting visual knowledge from web data. In *The IEEE Int. Conf. on Computer Vision (ICCV)*, December 2013.
2. Michelangelo Diligenti, Marco Gori, Marco Maggini, and Leonardo Rigutini. Bridging logic and kernel machines. *Machine learning*, 86(1):57–88, 2012.
3. Pedro F. Felzenszwalb and Daniel P. Huttenlocher. Efficient graph-based image segmentation. *International Journal of Computer Vision*, 59(2):167–181, 2004.
4. Donald Geman, Stuart Geman, Neil Hallonquist, and Laurent Younes. Visual turing test for computer vision systems. *Proceedings of the National Academy of Sciences*, 2015.
5. Giorgio Gnecco, Marco Gori, Stefano Melacci, and Marcello Sanguineti. Foundations of support constraint machines. *Neural computation*, 27(2):388–480, 2015.
6. Marco Gori. Semantic-based regularization and piaget's cognitive stages. *Neural Networks*, pages 1035–1036, 2009.
7. Marco Gori, Marco Lippi, Marco Maggini, and Stefano Melacci. On-line video motion estimation by invariant receptive inputs. In *CVPR workshops*, pages 712–717, 2014.
8. Marco Gori and Stefano Melacci. Constraint verification with kernel machines. *Neural Networks and Learning Systems, IEEE Transactions on*, 24(5):825–831, 2013.
9. Marco Gori, Stefano Melacci, Marco Lippi, and Marco Maggini. Information theoretic learning for pixel-based visual agents. In *ECCV*, pages 864–875. Springer, 2012.
10. Berthold K. Horn and Brian G. Schunck. Determining optical flow. In *1981 Technical Symposium East*, pages 319–331. International Society for Optics and Photonics, 1981.
11. Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in NIPS*, pages 1097–1105, 2012.
12. Ce Liu, Jenny Yuen, and Antonio Torralba. Nonparametric scene parsing via label transfer. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 33(12):2368–2382, 2011.
13. Mateusz Malinowski and Mario Fritz. Hard to cheat: A turing test based on answering questions about images. *CoRR*, abs/1501.03302, 2015.
14. Marcin Marszalek, Ivan Laptev, and Cordelia Schmid. Actions in context. In *CVPR*, pages 2929–2936. IEEE, 2009.
15. David C. McClelland. Testing for competence rather than for intelligence. *American psychologist*, 28(1):1, 1973.
16. Stefano Melacci and Mikhail Belkin. Laplacian Support Vector Machines Trained in the Primal. *Journal of Machine Learning Research*, 12:1149–1184, March 2011.
17. Stefano Melacci, Marco Lippi, Marco Gori, and Marco Maggini. Information-based learning of deep architectures for feature extraction. In *ICIAP*, pages 101–110. Springer, 2013.
18. Stefano Melacci, Marco Maggini, and Marco Gori. Semi–supervised learning with constraints for multi–view object recognition. In *ICANN*, pages 653–662. Springer, 2009.
19. Maxime Oquab, Leon Bottou, Ivan Laptev, and Josef Sivic. Learning and transferring mid-level image representations using convolutional neural networks. In *CVPR*, pages 1717–1724. IEEE, 2014.
20. Pierre Sermanet, David Eigen, Xiang Zhang, Michaël Mathieu, Rob Fergus, and Yann LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. *arXiv preprint arXiv:1312.6229*, 2013.
21. Lewis M Terman and Maude E Merrill. *Measuring intelligence.* ACC, 1961.
22. Joseph Tighe, Marc Niethammer, and Svetlana Lazebnik. Scene parsing with object instances and occlusion ordering. In *CVPR*, pages 3748–3755, 2014.
23. Antonio Torralba and Alexei A. Efros. Unbiased look at dataset bias. In *CVPR*, pages 1521–1528. IEEE, 2011.