BioVision: a Biomimetics Platform for Intrinsically Motivated Visual Saliency Learning

Céline Craye, David Filliat, Jean-François Goudou

BioVision: a Biomimetics Platform for Intrinsically Motivated Visual Saliency Learning

Céline Craye, David Filliat, and Jean-Fraçois Goudou

***Abstract*—We present BioVision, a bio-mimetics platform based on the human visual system. BioVision relies on the foveal vision principle based on a set of cameras with wide and narrow fields of view. We present in this platform a mechanism for learning visual saliency in an intrinsically motivated fashion. This model of saliency, learned and improved on-the-fly during the robot's exploration provides an efficient tool for localizing relevant objects within their environment.**

The proposed approach includes two intertwined components. On the one hand, a method for learning and incrementally updating a model of visual saliency from foveal observations. On the other hand, we investigate an autonomous exploration technique to efficiently learn such a saliency model. The proposed exploration, based on the IAC (*Intelligent Adaptive Curiosity*) algorithm is able to drive the robot's exploration so that samples selected by the robot are likely to improve the current model of saliency.

We then demonstrate that such a saliency model learned directly on a robot outperforms several state-of-the-art saliency techniques, and that IAC can drastically decrease the required time for learning a reliable saliency model. We also investigate the behavior of IAC in a non static environment, and how well this algorithm can adapt to changes.

***Index Terms*—Developmental robotics, visual saliency, convolutional neural networks, intelligent adaptive curiosity, incremental learning, bio-mimetics**

I. Introduction

Biological systems have always been a wide source of inspiration in robotics. In particular, the human vision system is a perfect example of biological system that has inspired number of applications related with computer vision. This work proposes to exploit mechanisms of the human visual attention, such as saliency to determine relevant targets in the visual field, and the foveal vision principle that provides a very high acuity in a restricted area of the field of view. In addition, the development of skills and knowledge by infants is actively studied in the developmental robotics community. Understanding and applying such mechanism on a robot could potentially lead to the new paradigm of conceiving robots able to learn with no or very limited human supervision. This topic is also a central element of this work.

In a more practical context, object localization in cluttered environments is still a difficult problem. Today, deep learning-based methods provide efficient ways to localize and identify a large set of objects in a wide variety of complex configurations [36], but they generally require hours or days of offline training, high GPU resources, thousand to millions of training images, and are not really flexible to novelty. Furthermore, a large variety of robots are meant to evolve essentially in restricted environments, interact with a limited amount of objects, to perform specific tasks. Thus, they do not require such wide scope capacity. On the other hand, they should be capable of some flexibility, being able to adapt to any novelty or change in their environment by quickly updating their representation. Learning to localize objects online and directly within the environment is then a very desirable property.

Nevertheless, online learning must come with a methodical exploration of the environment in order to gather relevant training samples. The displacement of the robot makes it possible to move to favorable observation conditions in order to improve recognition performances, but a critical point is to monitor this performance quality, and use this information to drive the robot accordingly.

In this article, we consider a foveated system, capable of visually exploring its environment to build a model of visual saliency enhancing objects of interest. Based upon the previous work [13], [15], we present a system able to:

- produce object-oriented visual saliency maps;
- learn the saliency model incrementally directly within the robot's environment;
- make the robot explore the environment autonomously and efficiently, by visiting in priority areas able to improve the saliency model.

More precisely, the system is composed with two major components. On the one hand, we present a method that exploits the foveal vision principle to learn a visual saliency model incrementally and without any user supervision. This model can be exploited to enhance objects of interest in the environment. On the other hand, we describe an implementation of the *Intelligent Adaptive Curiosity* (IAC) algorithm applied to the problem of saliency learning that drives the robot in its environment, so that learning is done is an efficient and organized manner. IAC then encapsulates the saliency learning technique and can be seen as a whole system for autonomous exploration and efficient learning. We demonstrate that our method for learning saliency online generates saliency maps that are more accurate than most state-of-the-art technique

Céline Craye is a PhD student at Thales SIX and at ENSTA Paristech, and a member of the INRIA FLOWERS team. Email: celine.craye@ensta-paristech.fr

David Filliat is a professor at ENSTA Paristech, and a member of the INRIA FLOWERS team. Email: david.filliat@ensta-paristech.fr

Jean-Fraçois Goudou is a R&I project manager at Thales SIX - Vision and Sensing laboratory. Email: jf.goudou@thalesgroup.com

U2IS, ENSTA ParisTech, Inria FLOWERS team, Université Paris-Saclay, 828 bd des Maréchaux, 91762 Palaiseau cedex France

Thales - SIX - Theresis - VisionLab 1, avenue Augustin Fresnel, 91767 Palaiseau, France

You can visit the project's repository at https://github.com/cececr/RL-IAC

in the robot's environment. In addition, the efficiency of IAC for exploration is evaluated with alternative environment exploration techniques, and the behavior of such algorithm is investigated when changes in the environment occurs.

In an earlier work [13], we have presented preliminary results on learning saliency in foveated platforms. However, the IAC algorithm was not applied yet for this kind of setups, and the evaluation was conducted on publicly available datasets rather than on the BioVision platform. In Craye *et al.* [15], we investigated the use of IAC on a mobile robot, where the goal was to displace the robots in different rooms of a building to efficiently learn visual saliency. We aim to demonstrate in this paper that the IAC mechanism is also applicable to a foveated system by taking advantage of the foveal and contextual views. We also present a new type of feature extractor based on convolutional neural networks, and present a batch of new experimentations to investigate the behavior of IAC in more details.

The article is organized as follows: in Section II, we mention some related work. Section III then presents the robotics platform of which most experiments are carried out. Section IV describes the method used to learn visual saliency incrementally, while Section V explains the exploration strategy based on IAC. We propose an experimental evaluation of our system in Section VI, and finally provide concluding remarks and perspectives in Section VII.

II. Related work

Our system is based on two major components, we then consider separately the related work on saliency maps and object localization, and the potential exploration strategies for improving knowledge about the environment. We last highlight our main contributions and positions towards state-of-the-art.

A. Visual attention and visual saliency

To efficiently analyze visual inputs and interact with objects in cluttered environments, robots often rely on a visual attention strategy. This mechanism turns the raw visual scene into selected and relevant information the robot should focus on. This concept has been widely studied and discussed [8], [29], [19], from biological and computer vision points of view. We restrict visual attention in this study to the localization of objects of interest.

Among robotics systems relying on visual attention, the *foveated systems* constitute an interesting study case, as they are designed based on bio-inpsired aspects. Foveated systems are typically composed with a *peripheral component* that aims to localize objects of interest, and a *foveal component*, used to get a high resolution representation of the target. Our robotics platform is typically classified among this type of systems. Bjorkman, Kragik *et al.* [3], [4], [34], [35], [50] have worked on a system of four cameras (two foveal, two peripheral). Not only were the pairs of cameras alternating between foveal and peripheral visions, but also estimating the depth of the environment from stereoscopy. Hueber *et al.* [27] have proposed a foveated platform based on foveal and contextual cameras for detecting and tracking moving targets. Other teams have simply used Pan-Tilt-Zoom (PTZ) cameras to have an easier setup based on a single camera, while alternative foveal and peripheral view points. Minut *et al.* [44] have used this kind of setup to jointly learn to localize and identify an object, while Kragic *et al.* [33] have used a descriptor to adjust the zoom level and enable a better recognition. Gould *et al.* [22] as well as Canas *et al.* [10] have mounted a PTZ camera on a mobile platform able to explore their environment by moving in a room and identifying simple objects.

Whether using a foveated system or not, visual attention is based on a pre-attentive stage, where potentially relevant targets are selected and uninformative areas are discarded, and an attentive stage, where more complex tasks (such as grasping [34] or object recognition [50]) are performed on the targets to obtain more information about them. This pre-attentive stage is typically related with the concept of visual saliency, defined as a 'subjective perceptual quality which makes some items in the world stand out from their neighbors and immediately grab our attention' [29]. The first computational models of visual attention were relying on saliency maps [30], representing the saliency of an image on a pixel-by-pixel basis. General convention is to associate a pixel intensity proportional to the pixel saliency.

Saliency maps can be either purely bottom-up [61], [17], [25], or refined by top-down modulation [23], [62], [18], [20]. Bottom-up saliency highlights stimuli that are intrinsically salient in their context, which may sometimes be sufficient for scene exploration [64]. However, top-down modulation, which highlights elements that are relevant for a specific task, is more meaningful for the problem of object detection in indoor environments. Saliency maps are either fixation-based [30], [17] or area-based [11], [20], [61]. Fixation-based approach is related with the probability of a human being to make a fixation at a given pixel position, while area-based approach consider salient elements (typically objects) as a whole area of the image. The latter approach is then closely related to object segmentation. In the context of a mobile robot in an indoor environment, our technique aims to build top-down, object-oriented models of saliency.

Machine learning, and especially deep learning have also been used for the generation of saliency maps. The best performance reported on saliency benchmarks [38], [37] is so far CNN-based. More interestingly, it was shown that CNN activation maps can be used as powerful objects detectors and trained on a weakly-supervised basis [63], [47]. For that reason, we investigate in this article the benefits of using CNN-based feature extractors in our technique.

B. Exploration strategies for learning

Visual attention in itself provides a potential exploration strategy. In this case a visual focus of interest is selected in the environment (from saliency maps computations for example), and the actions performed by the robot aim to provide more information about the selected target. When the robot is equipped with a foveated system [5], [58], the actions are typically saccades, so that the zoomed camera is oriented

towards informative regions. In the case of a mobile robot, actions are displacement to get a closer or better point of view of areas of interest [43], [33], [41], [7]. However, this kind of exploration supposes that a good model for selecting targets and reaching them is already available. Conversely, we are interested in this work about exploring for learning such visual attention capacities.

Exploration strategies for learning visual attention have been proposed for two kinds of applications. The first one [45], [18], [50] consists in refining a saliency map by finding appropriate weights to combine the extracted features. In particular, in the work of Rasoladeh *et al.* [50], the weights were refined online by alternating between bottom-up and top-down attentions. For that, a system of temporal differential equations was used to weight the importance of each component. The second type of methods aims to learn eye saccades in order to better perceive the objects of interest. These approaches typically rely on reinforcement learning techniques. Minut *et al.* [44] have presented a system based on a PTZ camera able to learn areas that most likely contains objects. Small saccades are also exploited to precisely localize and identify the object. Saccades can also be directed towards areas of interest within an object to improve its identification [49]. Lastly, Borji *et al.* [7] have proposed to learn a top-down attentional model able to simultaneously discriminate objects in an environment from their visual appearance, and learn a sequence of actions to reach a goal.

In the scope of developmental robotics, intrinsic motivations are also used as a drive for robot's acquisition of skills through experience and exploration. Intrinsic motivation, defined as a behavior driven by an intrinsic reward system (*i.e.* not related to an external goal, but to the acquisition of competences or knowledge), is a possible approach for guiding exploration in that regard. For example, Huang *et al.* [26] have used *novelty* to guide visual exploration, while Chentanez *et al.* [12] have used the error of prediction of *salient event* to speed up a classical reinforcement learning approach. To overcome limitations related to novelty or error in unlearnable situations, intrinsic motivation based on *progress* has been proposed [46], [1], [56]. The *Intelligent Adaptive Curiosity* (or IAC) [48] is one of the most emblematic implementation of intrinsically motivated exploration using progress. Learning progress has also been exploited in a reinforcement-learning context, typically with artificial curiosity [31], [55], or to make exploration flexible to changes in the environment or wrong assumptions [42].

Visual attention is an excellent study case for intrinsic motivation. In a general case, eye saccades and fixations can be seen as a way to actively sample information, and, as a result a form of intrinsically motivated exploration strategy [21], [2]. In neuroscience, several studies have highlighted the fact that eye movement patterns are not the same when trying to learn a skill, and once this skill is acquired [54]. In developmental robotics, visual attention is also considered as a common study case, most of the time used as a way to develop proprioceptive skills (for example, predict the position of the hand in a field of view [1]) or visual servoing skills (learning options from visual inputs [32], smooth pursuit [60], simultaneous gaze control and reaching [28]). However, learning the visual aspect of salient elements has not been examined so far in a developmental robotics framework, although clear evidences show that such saliency is, at least partially, learned [24], [6].

C. Contributions

So far, saliency maps are mostly used as black boxes and are not learned (although sometimes refined) directly during the exploration of a particular environment. Our first contribution is a method that incrementally learns saliency as the robot observes the environment. The produced saliency maps are therefore dedicated to the environment that was explored, but remain flexible to novelty. The model that is learned here is a top-down type of saliency, dedicated for generic robotics tasks (i.e. a saliency that detects objects the robot can interact with). The term saliency in this article is then more related to the concept of objectness, and the model that is learned is used to produce object-oriented saliency maps. Unlike most saliency techniques based on learning, ours is self-supervised, so that the robot is able to learn without any human annotation or assistance. The main mechanism consists in a transfer learning method between a weak object recognition in the fovea and the contextual view.

Our second contribution is the use of intrinsic motivation, and more precisely learning progress to drive the robot's exploration for the task of learning. To this end, we adapt the *Intelligent Adaptive Curiosity* (IAC) algorithm for the problem of saliency learning on a foveated platform.

A major difference between our approach and traditional visual attention systems is then the way we consider salient regions. Our purpose here is to consider saliency as something to learn rather that something to direct attention towards. We then direct our attention towards areas able to improve learning rather than towards salient elements themselves. Our focus of attention is therefore, in our system, at highly progressing areas.

III. The BioVision platform

The *BioVision platform* was developed in the scope of a bio-mimetics project, aiming to study and exhibit some aspects of the human vision. BioVision consists in a bio-mimetic head imitating the human vision system to develop new strategies for learning, recognition, or tracking. The platform first takes into account the properties of a human eye by relying on both foveal and contextual fields of view. Then, the visual cortex is modeled by deep convolutional neural networks performing object recognition. We use both the robotics platform and the object recognition system described in Section IV-B as tools for developing our algorithms.

BioVision is presented in Figure 1. It is composed of a single foveal camera, which is an EXG50 Baumer camera[1] with a narrow field of view of 5°. An Optotune EL-10-30TC liquid lens[2] is used on this camera to adjust the focus within

[1] http://www.baumer.com/us-en/products/identification-image-processing/industrial-cameras/

[2] http://www.optotune.com/technology/focus-tunable-lenses

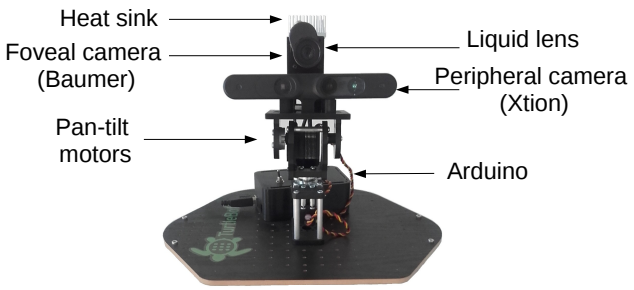

Fig. 1. The BioVision platform

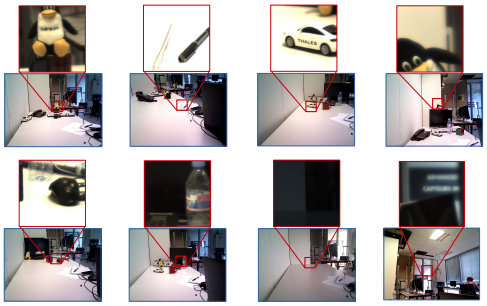

Fig. 2. Samples from foveal and contextual images

a few milliseconds. An RGB-D Asus Xtion Pro live camera[3] plays the role of the peripheral camera with a field of view of 57° horizontally. The pan-tilt motors are piezoelectric ones[4]. In this setup, the two cameras move along the same pan-tilt angles and are oriented the same way depending on the pan-tilt position. In addition, foveal and contextual cameras are calibrated so that an accurate position of the fovea's visual field can be mapped onto the peripheral one (See Figure 2).

IV. Incremental learning of visual saliency

This section describes the module able to learn a model of visual saliency from environment observations, and generates dedicated saliency maps. Figure 3 presents the general block architecture of the system. In a learning stage, the system extracts RGB features (see Section IV-A) from the peripheral stream and learns the visual aspect of salient elements within their context using an object classifier in the fovea as a supervision signal (see Section IV-B). Saliency learning is performed by a classifier (Section IV-C) that produces and constantly updates a saliency model. In an exploitation stage, the saliency model is used to generate environment specific saliency maps on the peripheral camera (Section IV-D).

An important aspect of our approach is that the learning signal is partial as only present in the fovea. Learning is also weakly supervised: saliency is estimated at the pixel level, while classification in the fovea provides a single label for the whole foveal area. By learning a model of saliency, we generate saliency maps estimating saliency on the whole frame. The online classifier is then able to generalize saliency over a weak and partial learning signal.

[3]https://www.asus.com/3D-Sensor/Xtion_PRO_LIVE/
[4]Pan: ServoCity DDP125. Tilt: ServoCity DDP500

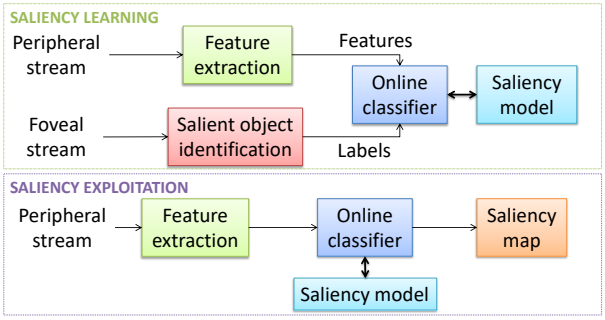

Fig. 3. General architecture of the saliency learning algorithm

A. Feature extraction

In this work, feature extraction is based on convolutional neural networks and is fully independant from the classification step. Considering deep learning frameworks, an end-to-end neural network architecture may be used for saliency learning, starting with convolutional layers and ending up with fully connected one. However, meta-parameters (learning rate, minibatch size, *etc.*) are hard to configure to allow an efficient incremental learning. A bad configuration could significantly deteriorate weights that were correctly learned for another problem. We then consider another approach to exploit deep neural networks: the use of the first layers of a well-trained network as a feature extractor. This way, we avoid instability problems, and the module is easily plugged in our architecture.

We base our feature extraction upon the ideas of Zhou *et al.* [63]. In their article, a GoogLeNet architecture is used and fine tuned to perform object localization. The end of the network is replaced by a global average pooling layer, followed by fully connected layers providing strong localization capacities and trained on a weakly supervised dataset. This type or architecture is then able to produce, in some sense, class specific saliency maps. In addition, the weights of this network are publicly available.

We then use this available trained model and do not consider the layers after the global average pooling one. The feature extraction is done at the level of the *class activation mapping*, or CAM layer (called *CAM-CONV* in the network). This corresponds to the last fully convolutional layer of the network. According to Zhou *et al.*, this layer is the one at which highly discriminative areas are enhanced.

To get a set of feature maps, we then feed the network with the original RGB input image, without resizing it, and extract the 1024 maps of the CAM layer (See Figure 4).

Because of striding and pooling in the network, the output feature maps have a resolution that is 16 times lower than the input image. To overcome this loss of resolution, we present in Section IV-D a method to reconstruct saliency maps at the original scale.

B. Salient object identification

To learn a model of saliency, we base our method upon an object classifier applied to the foveal stream, playing the role of a learning signal. This learning signal has the property

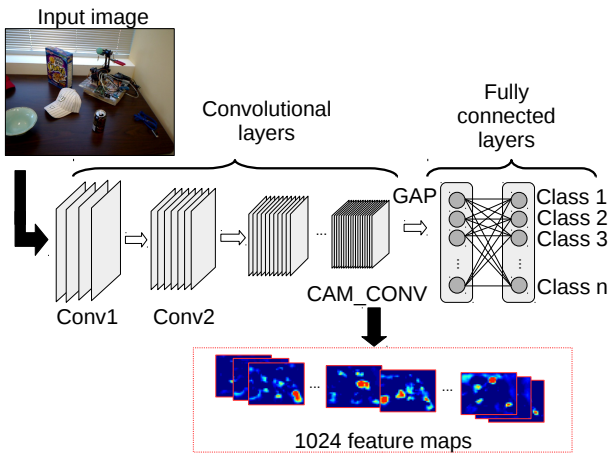

Fig. 4. Example of feature maps extraction from an input RGB image

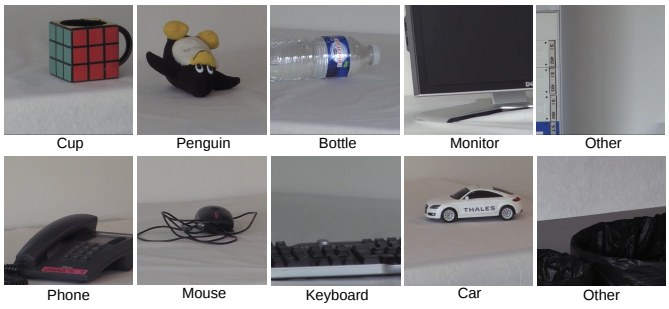

Fig. 5. Sample images from the training dataset of the 9-classes object classifier

of being reliable, but only provides a partial and weakly supervised information. The goal is then to use the result of the object classifier as a local estimation of the saliency, and to transfer this information to the peripheral view. This way, the learning signal is generalized to produce saliency maps in the contextual view.

To generate the learning signal, we consider an object classifier based on convolutional neural networks, and more precisely, an Alexnet [36] architecture pre-trained on the ImageNet dataset [52]. We replaced the last layer of the network to identify 8 different categories of objects, and a last class called '*other*', containing any other types of elements.

To train the network, we constituted a dataset of objects captured from the foveal camera. In total, 8 different objects were placed on a white table and recorded at thousands of different points of view (see Figure 5). The additional class '*other*' was collected similarly with various kinds of background surrounding the objects. In total, around 10 000 images were collected this way. The network was finally fine-tuned with the collected dataset, and an accuracy of 99 % was measured on the validation set.

To exploit this classifier as a learning signal for saliency learning, we take this classifier already trained, and use it to infer in real time the class of the object observed by the fovea. This output is then converted into a saliency label, where the eight objects are considered to be salient (annotated with the '*salient*' label), and any other visual item as being '*not salient*'. Therefore, if the output of the classifier is the class '*other*', the fovea is annotated '*not salient*'.

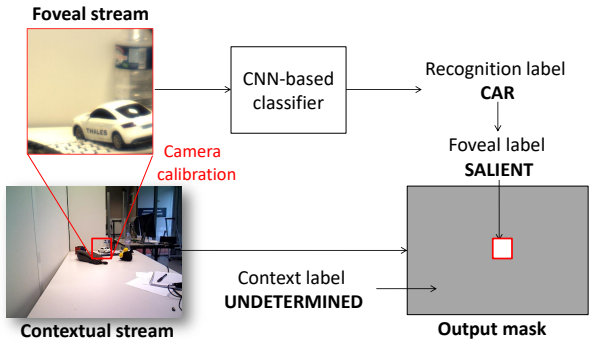

Fig. 6. Segmentation mask from foveal object recognition

In a second stage, we convert this learning signal into a mask of the same size as the peripheral camera, on which saliency should be learned. For that, we rely on the extrinsic calibration of the foveal and peripheral cameras, to precisely determine the boundaries of the fovea in the peripheral field of view. Then, the mask, having the same dimensions as the peripheral images, is such that the label determined in the fovea ('*salient*' or '*not salient*') is attributed to each pixel of the fovea's boundaries on the peripheral frame. Each pixel outside the fovea is then labeled '*undetermined*'. Figure 6 illustrates the general approach to transfer the result of classification in the fovea to a segmentation mask.

C. Online learning

Learning is made possible by using the feature maps, the segmentation mask as a learning signal, and an online classifier. The classifier is continuously updated based on the foveal and contextual observations, turned into a set of labels and features: the segmentation mask is first resized to the same size as the feature maps. For each pixel, the 1024 associated features are collected and turned into a feature vector so that a sample of features-label is associated for each of them. These samples constitute the dataset send to our classifier to train our saliency model. Only pixels of the fovea are selected to feed the saliency model.

The classifier used in our implementation is an online version on random forests. Random forests are natively not designed for online training, although a few incremental versions have been proposed [53], [40]. Nevertheless, none of these online versions was satisfying in terms of speed and performance, so we adapted the offline version of random forests to make re-training fast enough. For each new frame, we consider the annotated pixels, and add the corresponding data to a dataset cumulated from the beginning of the sequence. Then, we update the classifier by only re-training a small fraction of the forest at a time: we randomly select 4 trees among the 50 in the forest, and we retrain those tree with 70% of the dataset cumulated from the beginning of the sequence. Lastly we restrict the size of the cumulated dataset to 100000 samples. If the dataset exceeds this size, we randomly remove samples to meet the maximum size requirement. As a result, after each

update, the classifier is able to estimate the saliency of an input based on the model trained with the previous observations, and the peripheral image only. More implementation details can be found in [16].

D. Saliency map reconstruction

Saliency maps are generated by applying the classifier to the peripheral images. To this end, features are extracted from an input image and are sent to the classifier to produce a saliency evaluation. For each pixel of the feature map, the classifier outputs a probability of the pixel to be salient. The output score is then a value between 0 ('*not salient*') and 1('*salient*') corresponding to a fuzzy state of saliency (this fuzzy state is represented as grayscale or heatmaps in the experimental results). To rescale this low-resolution saliency map to the original input image size (recall that the deep feature extraction downsamples the image by a factor of 16), we generate SEEDS superpixels [57] from the original image (350 superpixels for 640×480 images in our experiments). We associate a low-resolution pixel to each superpixel by finding the pixel that would be the closest to the superpixel centroid if rescaled at the same size. The saliency value estimated for this pixel is then used to cover the entire superpixel.

V. Intrinsically Motivated Exploration

A. Mechanisms

Intrinsically motivated exploration is done by using the Intelligent Adaptive Curiosity (IAC) algorithm. IAC can be seen as a way to guide the robot's actions when exploration is dedicated to the task of learning something. Unlike exploration strategies whose aim is to have an extensive coverage of the exploration space, IAC focuses on particular areas of the space so that learning is done efficiently. The essential component of this technique is a local measure of the learning progress, that catches the attention of the robot until knowledge has been acquired. We here focus on describing the mechanism in our particular framework.

IAC is constituted of four main components:

- a *learner* that learns a particular model;
- a way to divide the exploration space into *regions*;
- a *meta-learner* that monitors the learning evolution in each region and estimates progresses[5];
- a *policy* that determines the next action to take given the state of progress in each region.

Figure 7 summarizes the main components of the architecture and the way they interact with each other. We here provide a brief recall on the basic procedure for using it. At time t, the robot receives observations from the foveal and peripheral streams. They are, on the one hand, turned into a segmentation mask S_t, and, on the other hand, sent to the online classifier to estimate the saliency map $\widetilde{S}_t$. Then, the learner and meta-learner are both updated: S_t is first turned into labels and sent with the extracted features F_t to the online classifier (learner)

[5]For more simplicity, we consider the meta-learner and the knowledge gain assessor [48] as a unique module, both monitoring error and deriving progress.

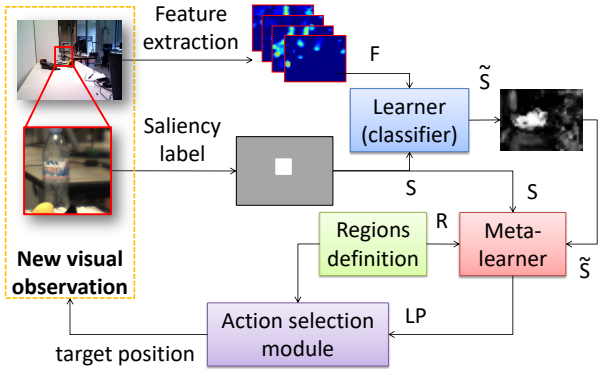

Fig. 7. General architecture of the IAC algorithm for saliency learning.

for a model update. Second, as this observation was taken at a particular point of view, in a region R, the prediction error $\|S_t - \widetilde{S}_t\|$ is added to the error history of R, and the progress in region R is re-evaluated. Lastly, the next action of the robot is calculated from a policy depending on the learning progress. The robot then takes the action, ends up in a new position and receives an new visual input that is to be processed, and so on.

B. Learner

In our implementation, the learner is the online classifier used in the saliency learning technique of Section IV. More formally, our learner tries to construct a prediction function $M : I \rightarrow \widetilde{S}$, able to estimate the saliency $\widetilde{S}$ of the visual field given an input RGB image I. This process is done by moving in the environment, collecting observations after each displacement, and updating the learner's model M after each observation.

After an observation at time t, features $F_I(t)$ are extracted from the peripheral stream, and the segmentation mask $S(t)$ is derived from the fovea, playing the role of labels for the learner. The features are sent to the learner to infer the saliency map $\widetilde{S(t)}$ from M_{t-1}. This saliency map is used by the meta-learning module later on for learning quality estimation (see Section V-D). Then, both $F_I(t)$ and $S(t)$ are used to update the learner's model M_t.

C. Regions definition

In our experiments BioVision is placed on a table. To modify the point of view, a pan or tilt motor command is used so as to change the inclination of the cameras.

Positions of the reachable points of view are then defined as a pan and tilt position (p, t). Regions are defined by cutting the pan and tilt spaces in regular intervals. The robot is then in region R if the pan tilt position falls within the boundaries of this region. Figure 8 illustrates the regions boundaries and makes the link between the pan-tilt axes and the foveal point of view.

Of course, pan and tilt positions are not so easily convertible to a foveal point of view. However, we make the assumption that objects are far enough to consider the optical center of the foveal camera to be at the interserction of the pan and tilt

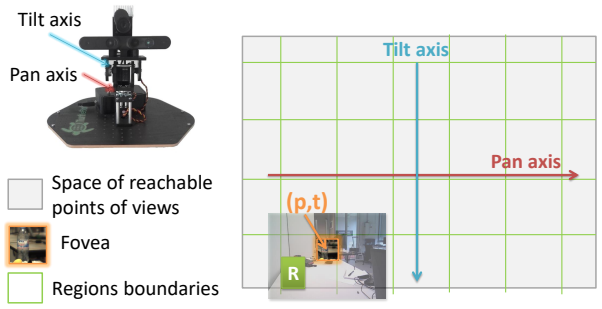

Fig. 8. Regions definition in the pan-tilt space

axes. Pan and tilt values are easy to obtain and actions are simple pan and tilt motor commands. Our second assumption is to consider each displacement command to fall in the right position which, given the precision of the motor, is realistic.

In practice, we divide the pan positions (ranging from -60° to 60°) into 4 intervals and the tilt (ranging from -30° to 30°) into 4 intervals as well. We then have the space divided into 16 regions.

D. Meta-learner

The meta-learner aims to monitor the local error made by the learner, and derive an estimate of the learning progress. The local estimation is made possible by grouping and making statistics on samples collected within the same region. Recall that the robot is in region R_i at time t if its current position falls within R_i's boundaries. We provide in this section a method for estimating learning progress.

The error made by the learner at time t is computed by comparing the segmentation mask $S(t)$ with the corresponding saliency map $\widetilde{S}(t)$. For that, we consider the *image labels set* $L(t)$ by keeping only pixels labeled '*salient*' or '*not salient*' in $S(t)$. We binarizefootnoteRecall that each pixel of $\widetilde{S}(t)$ is a probability of being salient. We binarize these probabilities by a threshold at 0.5 the saliency map $\widetilde{S}(t)$ for each of these pixels to obtain the *image estimation set* $E(t)$Ł astly, we compute the corresponding estimated error $Err(t)$ based on Equation 1:

$$Err(t) = 1 - F_1(L(t), E(t)) \tag{1}$$

where $F_1(.,.)$ is the *F1 score*:

$$F_1 = 2 \cdot \frac{precision \cdot recall}{precision + recall} = \frac{2tp}{2tp + fp + fn} \tag{2}$$

with tp, fp and fn the true positives, false positives and false negatives[6].

To evaluate learning progress, the meta-learner stores a history of the prediction error for each region. Suppose now that at time t, the robot is in region R_i and makes an observation in this region. Suppose that in this region, $n-1$ observations were already recorded and added to the history from the beginning of the experiment. The observation at time t is then the nth of region R_i, and the learning error $Err_i(n)$ associated with this observation is then added to the history of R_i.

The estimation of the learning progress in R_i, is obtained by exploiting the error history sequence. For that, we apply a linear regression of the error history over the last τ samples:

$$\begin{pmatrix} Err_i(n-\tau) \\ \vdots \\ Err_i(n) \end{pmatrix} = \beta_i(n) \times \begin{pmatrix} n-\tau \\ \vdots \\ n \end{pmatrix} + \begin{pmatrix} \epsilon(n-\tau) \\ \vdots \\ \epsilon(n) \end{pmatrix} \tag{3}$$

with $\epsilon(n)$ the residual error and $\beta_i(n)$ the regression coefficient. $\beta_i(n)$ then represents the derivative of the learning error after n observations. The learning progress being defined as the derivative of the learning curve (opposite of the prediction error), we obtain the progress $LP_i(n)$ in region R_i by Equation 4:

$$LP_i(n) = \frac{2}{\pi} |atan(-\beta_i(n))| \tag{4}$$

We transform the slope β_i with an arctangent to have the learning progress normalized between -1 and 1, and we consider the absolute value of the arctangent to force the robot to explore regions where learning is decreasing as well.

E. Action policy

In most IAC implementations, learning progress is used directly as an intrinsic reward the robot should follow. More precisely, many implementations select the region having the highest learning progress and randomly choose an action among all possible actions leading to that region. As the next action is selected based on immediate learning progress, without any long term planning consideration, we call this behavior *greedy*.

We also follow an ϵ-greedy procedure to select the next action. However, we do not directly select the most progressing region, but select the region with a probability proportional to the learning progress. This idea was already proposed by Baranes *et al.* [1], and suggests that the probability of the next region to visit r being region R_i is given by equation 5

$$Pr(r = R_i) \begin{cases} \dfrac{LP_i(t)}{\sum_{j=1}^{N} LP_j(t)} & \text{if } v > \epsilon \\ \dfrac{1}{N} & \text{otherwise} \end{cases} \tag{5}$$

where $LP_i(t)$ is the learning progress, v is a uniform random variable, and $\epsilon = 0.25$ is used to select a random region 25% of the time.

Once the next region is selected, a position within this region is randomly determined as the target to reach. The corresponding action to reach this position is then used to displace the robot.

VI. Experimental results

A. Experimental setup

To demonstrate the efficiency of our approach, we carry out experiments on the BioVision platform, as well as on a

[6]We use the F_1 score as our error metrics, because '*not salient*' pixels are representing more than 90% of the samples, making accuracy inappropriate for error estimation.

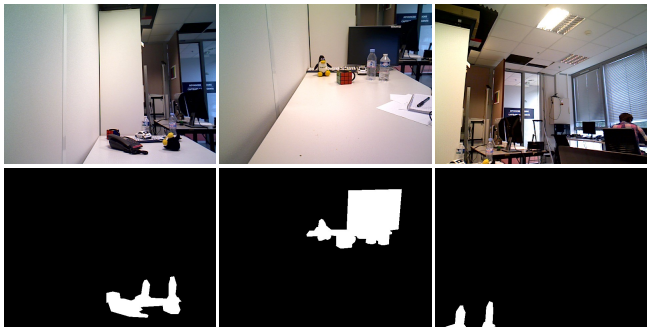

Fig. 9. Contextual images (top) and associated ground truth mask (bottom)

publicly available database. We describe in this section the main features of our datasets.

During the sequence acquisition, BioVision was put at the extremity of a two meters-length table, looking mainly at objects at the other end. The table and adjacent wall are all white, so that objects on the table are likely to be naturally salient in this local context. However, the background of the room is largely visible by BioVision, and contains a lot of distractors. To make the experiment consistent with the object classifier described in Section IV-B, we put the same list of objects on the table, sometimes with additional distractors (such as pens).

Our dataset is composed with 10 sequences recorded at different times of the day. For each sequence, we placed an arbitrary configuration of objects on the table, either known by the object classifier or not. We then let BioVision observe the environment for about three minutes, meaning that the robot was randomly selecting a pan-tilt position, reaching this position, acquiring both contextual and foveal images at this position, and selecting a new one. In total, around 2000 *observations* were collected this way (see a few examples in Figure 2.

To evaluate our results from this dataset, we constituted a training and a testing set. The training set is composed with the observations taken from 7 out of the 10 sequences, and the testing set is composed with the 3 last sequences. In those three sequences, we randomly selected 150 samples that we annotated. To do so, we manually segmented the contextual images of the selected samples, to create ground truth masks of salient and non salient elements. Note that in this context, the only salient elements that were manually segmented are the one placed on the table, even if some elements present in the background (such as computers and keyboards) could also have been annotated as salient elements. Figure 9 shows an example of contextual images and associated ground truth masks.

To demonstrate that our saliency method is also efficient on other types of data, we performed experiments on a public dataset. We chose for that the *Washington dataset*, and more precisely the *RGB-D scenes dataset* [39]. This dataset is composed with 8 video sequences of indoor scenes with everyday-life objects placed on tabletops. In total, around 1500 RGB-D frames are available in this dataset along with bounding boxes around objects. As this dataset is not providing any foveal images, we cannot use the object classifier in the fovea as a learning signal. We instead use an object detector based on the depth-map to this end. This depth-based object detection has been described in previous publications that one could refer to for more details [14], [15]. Apart from this, the saliency learning process is exactly the same.

B. Incremental saliency learning

Our first evaluations are related with the mechanism of incremental saliency learning, without considering any exploration aspect. We here provide quantitative and qualitative results to demonstrate the efficiency as well as the generalization capacity of the classifier. In this section saliency maps are represented either as a grayscale image (black standing for '*not salient*' and white for '*salient*'), or as a heatmap for which red are the most salient areas and blue are the least salient ones.

1) Saliency evolution: Figure 10 illustrates the evolution of the saliency in time for a fixed image. In this setup, the robot is learning by making random observations in the exploration space and updating the model after each of them. After each model update, we re-estimate the saliency of a given frame, that the robot does not see during learning. We display in Figure 10 a few of these saliency maps. The first observation is made with the fovea centered at the penguin's face, which is the only salient area of the saliency map at this time. Then, as the system takes additional observations, the saliency map is progressively refined. For a better understanding of the process, a video illustrating this mechanism is available [7].Regarding result on Figure 10, the experiment is done so that BioVision's head is scanning sequentially the field of view. This means that at frame 1, BioVision looks at the bottom left corner of the scene, and only perceives the penguin as being salient. At the end of the experiment, BioVision is looking at the top right hand corner of the scene and has gone through all the salient objects. However, this sequential scan is not a good strategy for learning, as the system tends to forget the first seen elements (typically the penguin). For that reason, a wise exploration strategy such as IAC is of paramount importance.

2) Generalization from foveal observation: To demonstrate the generalization capability from foveal observations, we train and evaluate the model on the same sequence. In Figure 11, we display a sample frame (cropped around salient elements of the scene) and some foveal observations used to feed the saliency model. All of these observations (and therefore, each pixels within the fovea) were labeled '*salient*' by the object recognizer, while it is clear that a large fraction of the pixels are not part of salient objects. Nevertheless, when looking at the resulting saliency map, the classifier was able to discriminate areas that were part of the objects, and areas from the background (walls behind or table). Of course, the discrimination process is not perfect: elements such as plastic bottles were not clearly learned as salient, and portions of the wall behind objects were partially classified salient, but the weak supervision can indeed lead to more precise saliency maps.

[7]https://github.com/cececr/RL-IAC

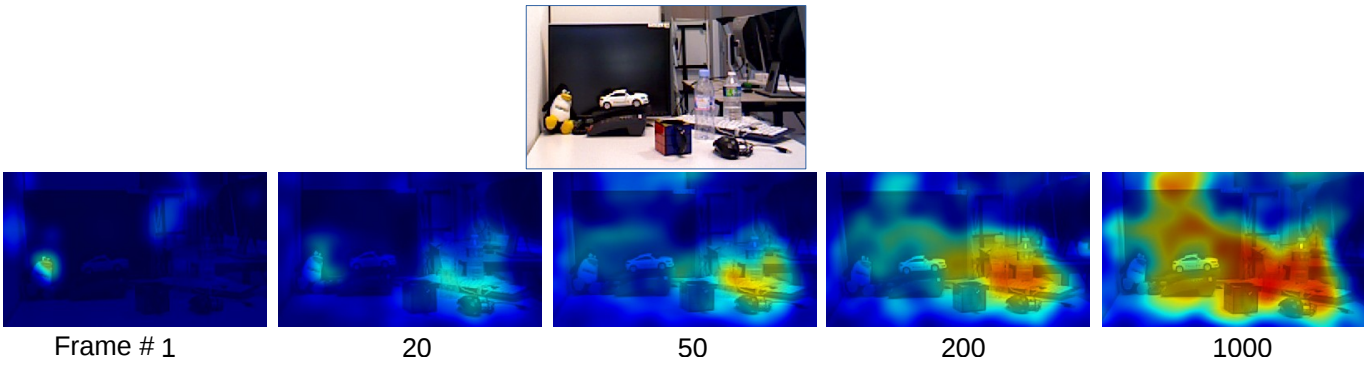

Fig. 10. Evolution of the saliency by applying the model on a fixed image, while the model is improved by observation taken from different images.

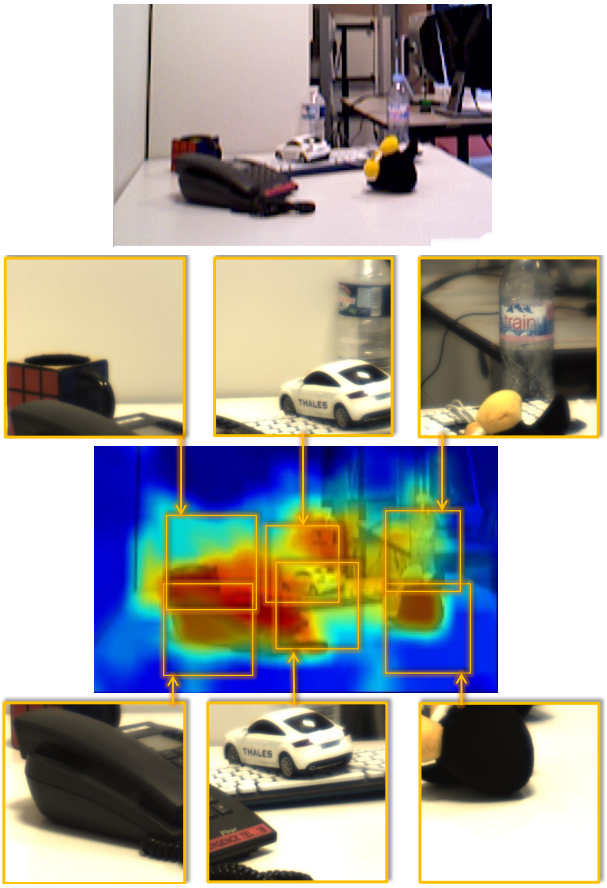

Fig. 11. Generalization capabilities from a weakly supervised signal (the foveal image). All the orange rectangles were labeled Salient after foveal object recognition, but the classifier is able to refine this labeling to provide better object segmentation.

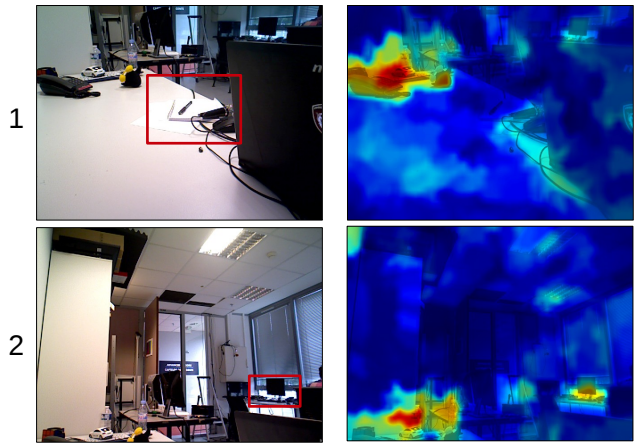

Fig. 12. Some interesting properties of the saliency model: Naturally salient distractors (such as pen in sample 1) are classified not salient, and areas containing salient elements (such as keyboard in sample 2) are highlighted.

In figure 12, we point out two additional features of the saliency model. First, distractors such as the pen (sample 1) that would be naturally salient are correctly classified not salient. Second, the classifier is able to retrieve similarities in areas that have not been observed by the fovea. In sample 2, a computer keyboard and monitor are detected at the other side of the room.

3) Comparison with state-of-the-art: To evaluate the saliency model versus existing state-of-the-art approaches, we analyze the final performance reached by the classifier when all samples of the training set are used. We denote in this section our incremental saliency learning approach and produced saliency maps as ISL.

We measure the saliency performance based on the ROC curve evaluation. The ROC (receiver operating characteristic) curve is among the most common techniques to evaluate saliency [51]. The idea is to construct a curve representing the true positive rate versus the false positive rate of the method. Theoretically, a method having a higher area under the ROC curve should be more efficient. To demonstrate the accuracy of our saliency model, we compute the ROC curves on the two presented datasets. The results were obtained by training ISL on their associated training set, and evaluating on their evaluation set.

For comparison, we selected three publicly available saliency algorithms and computed the ROC curves for each method on each of the three datasets. First, **BMS** [61] is among the most accurate RGB saliency methods according to the MIT *saliency benchmark* [9]. We use BMS with the configuration that highlights salient objects rather than salient fixations. Second, we use the new version of the **VOCUS2** algorithm [20] along with the configuration file dedicated to the task of object detection in cluttered scenes (top-down saliency). Third, we compare our method with saliency maps

produced with the **CAM** [63] model. This model is trained to detect objects among the 1000 classes of ILSVRC. For a fair comparison, we disabled classes that were not present in the images of our datasets (i.e. their output score were systematically set to 0), so that the produced saliency maps were responsive to relevant objects only. In addition, the maps produced by the CAM approach have the same low resolution than our model. We therefore apply the superpixels approach presented in Section IV-D to increase the resolution of these maps. To evaluate our feature extractor versus the one proposed in previous work [15], we generate saliency maps from both the CNN-based feature extractor (denoted as **ISL** here), and the former feature extractor (denoted as **ISL-Make3D**) that was used in [15].

In figure 13, a visual comparison between the methods is presented. Samples 1 and 2 are from the BioVision dataset while the three other samples are from the RGB-D scenes dataset. On the BioVision dataset, the background is composed with a white table and walls that are not naturally salient, but also with a highly textured background at the other end of the room (in sample 2 for example). The three state-of-the-art techniques tend to be very responsive to this highly textured background, and consider these areas as being salient. Conversely ISL and ISL-Make3D have learned that these specific textures were part of the background, and are then able to classify them correctly. The superpixel reconstruction approach makes it possible to retrieve shapes of salient objects (in spite of the low-resolution feature maps produced by the output of the CNN). When applied to the CAM saliency map, the superpixel reconstruction does not provide such good results. This might be because the produced saliency is much more diffuse (as a comparison, the CAM algorithm results are displayed without superpixel reconstruction for samples 1, 2 and 3). In addition ISL-Make3D produces saliency maps with finer details than ISL (see for example the case of sample 4), but is constructed with features that have a weaker generalization capacity than the deep features used in ISL (for example, a shadowed portion of the wall in sample 1 is found to be salient by ISL-Make3D).

The numerical results displayed in Figure 14 present the ROC curves of the five evaluated techniques on both the BioVision and the RGB-D scenes datasets. These results suggest that ISL outperforms all other techniques. The performance of ISL-Make3D is drastically different on the two datasets, because of the generalization capacity of the extracted features. On the BioVision dataset, the information extracted from the fovea only represent a very small fraction of the image, whereas the depth-based segmentation process used on the RGB-D scenes dataset provides an estimation of the saliency on a much wider area of the image.

C. Intrinsically motivated exploration

To evaluate the benefits of IAC as a guide for exploration, we conducted our experiments in semi-simulated environments, using the recorded sequences on the BioVision platform. We called these experiments semi-simulated as saliency was learned from real images taken from these sequences, but actions taken by the robot were simulated.

More precisely, to make the robot reach a position in our setup, we simply select one of the frames recorded in the sequence. This process has in practice no physical cost and do not take into account the time required to take this action in real life. Once selected, we just consider that the robot has reached this position and can start processing the associated observation.

For BioVision, regions were defined before the experiment by dividing the reachable pan-tilt positions into 4 pan intervals and 4 tilt intervals (for a total of 16 regions). The size of each region was defined so that regions were not equally interesting. Some of them only contained white wall, without any salient object, while other are mainly composed with them. Figure 15 illustrate a sample of foveal images reachable in each region.

For *RGB-D scenes*, we used the sequence *table small 2* only. The video sequences are provided without any localization information. However, the trajectory of the acquisition sensor is such that each point of view is seen only once in the sequence. We then created 5 regions by dividing the video into five sub-sequences of equal length.

1) IAC versus other exploration techniques: To demonstrate the efficiency of IAC versus other exploration techniques, we ran a set of experiments for three different types of exploration. The first consists in following the order of the recorded sequence to determine the next region to visit. We call this exploration strategy **chronological**. The second consists in selecting a position randomly among all possible positions, without considering any progress measure. We call it **random** exploration. Last, we evaluate the performance of **IAC**. To get a better visualization of the performance, we also display the score that the static bottom-up method called **BMS** (see Section VI-B3) obtains on the evaluation set, as well as an **offline** training with all the frames of the sequence at the same time. Those results are reported in Figure 16.

Each exploration strategy was tested 10 times on each dataset and results are reported based on the average and variance over those experiments. The performance of the system was evaluated using the evolution of the *overall error rate* of the system: based on the reference frames on which a ground truth is available, we compare the estimated saliency map for all of these frames with the available ground truth. We then use the formula provided by Equation 1 on each frame and take the average error.

Given this measure, we can monitor the evolution of the error rate all along the experiment and objectively compare the different exploration approaches. Note that the *overall error rate* has a different signification than the *regional error rate* of Equation 1 used to compute learning progress. Indeed, the *regional error rate* is an intrinsic metrics, whose evaluation is based on segmentation rather than on an external ground truth. Second, we use the F1 score, rather than the ROC curve for evaluation to be consistent with our evaluation of the learning progress.

As expected, the offline model is a lower bound of the error rate, and the system tends to reach this limit after a certain number of observations, whatever the exploration strategy. Our method rapidly outperforms BMS when enough observations are obtained. The chronological exploration is

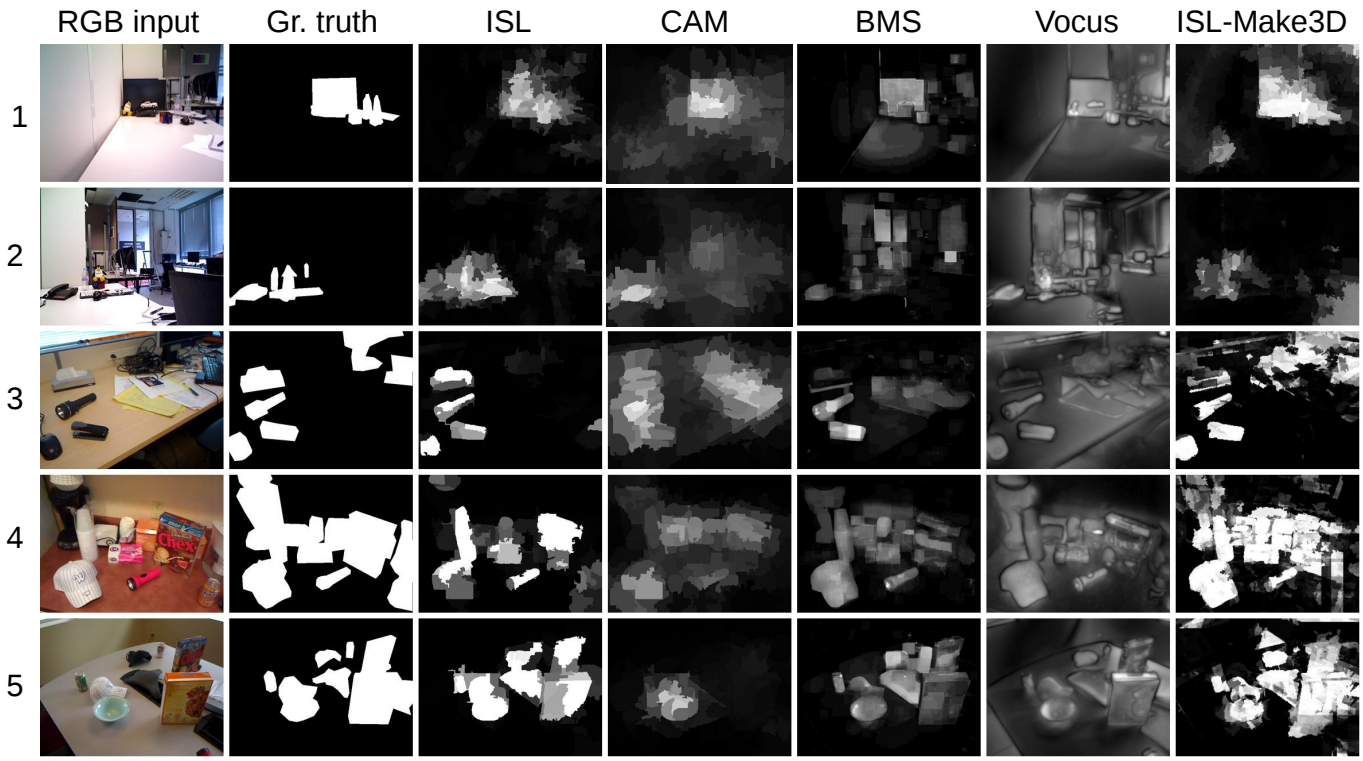

Fig. 13. Sample results of saliency maps compared with state-of-the-art bottom-up methods, obtained on both the BioVision and the RGB-D scenes datasets.

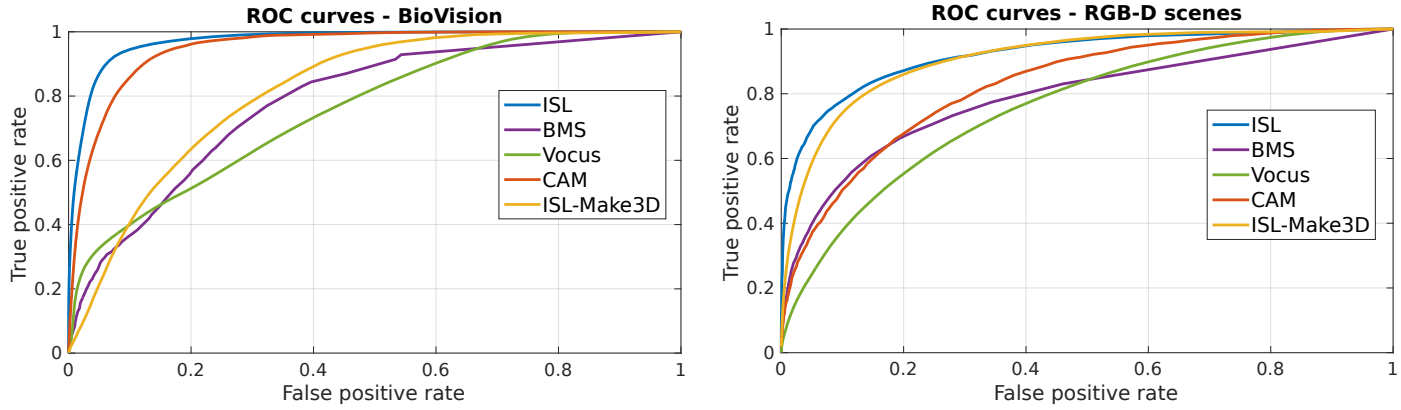

Fig. 14. ROC curves comparing state-of-the-art approaches on two datasets

slower to converge than random exploration at the beginning. Lastly, IAC seems to be the exploration strategy for which learning is the fastest. Note that the difference between random and IAC-based exploration is less significant on the BioVision sequence, but the error variance of IAC is much lower than the one for random exploration.

2) Adaptation to a change in the environment: We now study the evolution of the exploration in the case where the environment is changing. Suppose now that BioVision is learning the saliency of the environment, when someone comes and suddenly moves the positions of the objects on the table. The learning curve in the regions containing objects should be drastically modified, and the exploration strategy should exhibit different patterns.

To produce such behavior, we consider three sequences of the BioVision dataset: we make BioVision start learning with a certain sequence, and we switch to a new one during the experiment. We use for that a first sequence with a moderate number of objects, and switch with two different sequences: first, a sequence in which all objects were removed from the table (called **no objects** in the explanations). Second, a sequence where additional objects were put on the table(**new objects** in the explanations). Figure 17 illustrates the switch principle to simulate the change in the environment. To further analyze the behavior of IAC in this configuration, we run several experiments by changing the time at which the switch operates: after 100 or 400 observations, or never.

We first present in Figure 18 the evolution of the learning curve when operating a switch in the environment after 100 and 400 observations (curves **100** and **400** in the figure). We

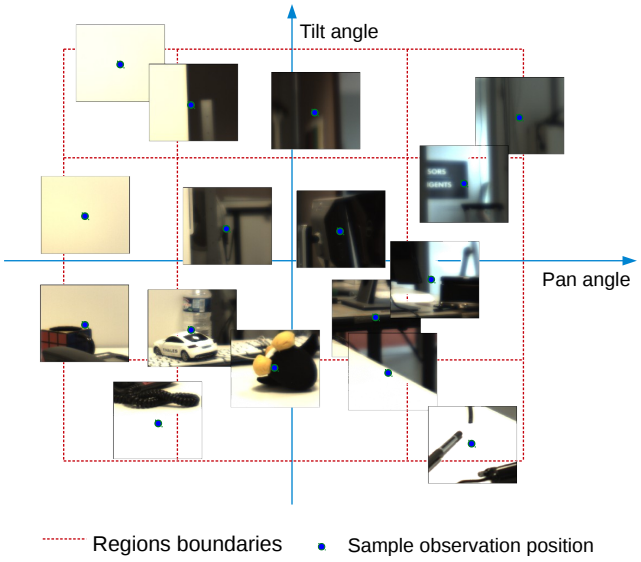

Fig. 15. BioVision's 16 regions and associated sample foveal images.

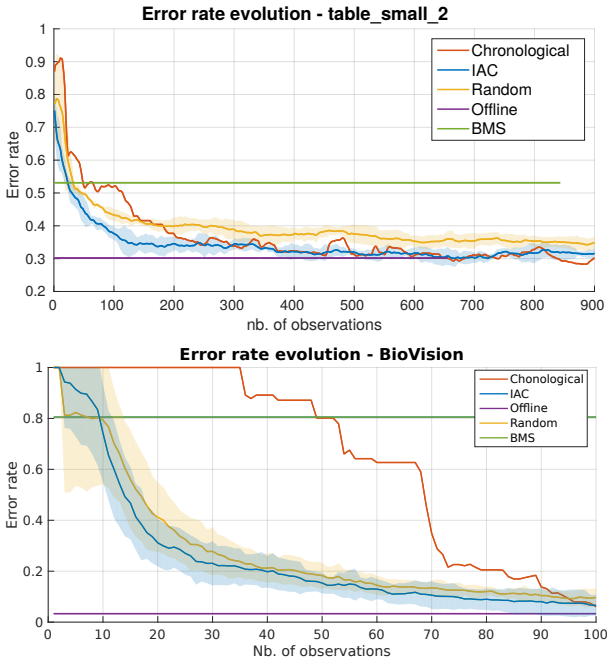

Fig. 16. Evolution of the error rate for several exploration strategies

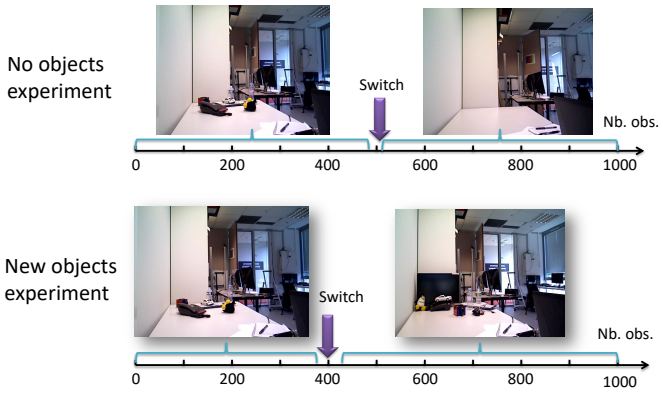

Fig. 17. Switch between datasets to simulate a change in the environment of the robot

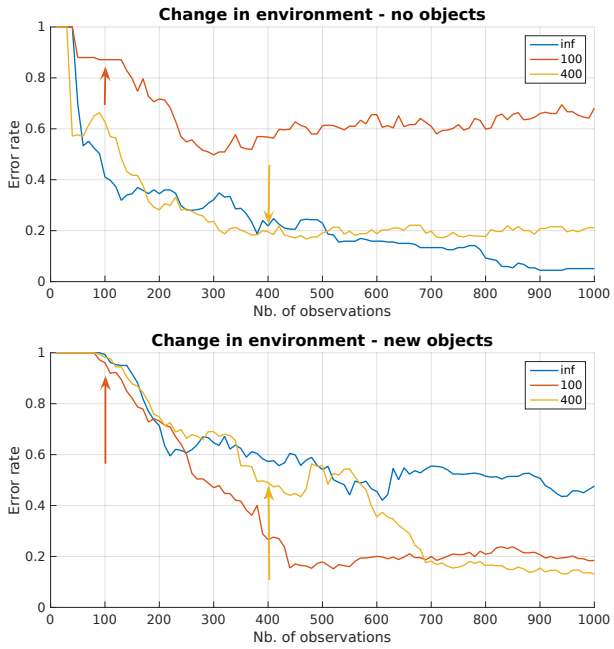

Fig. 18. Error rate evolution when modifying the environment of the robot. The red and yellow arrows point out the frame at which the change in the environment occurs.

also display the resulting error rate without any change (curve **inf**). In the experiments, the error rate is evaluated based on the ground truth of the sequences constituting each experiment. As a result, the **no objects** and **new objects** experiments are not evaluated on the same set of ground truth and do not have the same error rate curves. As expected, the change in the environment (indicated by arrows in the plots) strongly modifies the learning curve quality: when the environment switches to a table without any object, the scene is simplified, and the robot cannot get additional samples to learn the visual aspect of salient objects. As a result, the later the switch occurs, the better the learning. This is verified by the first plot of the figure, where the switch after 100 observations produces a very bad model. Conversely, when the new configuration of the environment is a more complex one, the robot is able to learn the aspect of additional objects, which is likely to improve the learning quality. This time, the switch is supposed to make the model better. This is verified by the second plot of the figure, where the switch at 100 observations enables a fast decrease of the error rate. However, when the switch never occurs (represented by the **inf** curve), the error rate cannot reach a descent performance. Lastly, the curve corresponding to a switch at 400 observations seems to have the best final performance (as compared to the switch at 100). This results would need to be confirmed with additional experiments, but this may be due to the greater number of samples from the first configuration that produces a better balanced model.

For these same experiments, we represent the amount of time spent in each region. To better analyze the results, we

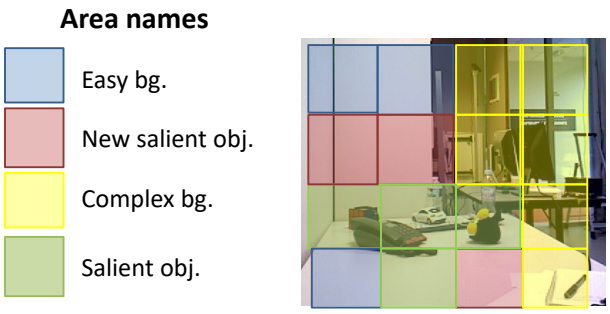

Fig. 19. The 16 regions are classified according to 4 groups (of a different color). Recall that these regions were defined by dividing the pan-tilt space into 16 equally-spaced squares. Their approximative positions are projected on this image as an example.

group the 16 regions in four *areas*: first, regions containing salient objects at the beginning of the experiment. Second, the one containing a background easy to learn (white walls and table). Third, the one containing a complex background. Lastly, the one for which the more complex environment contains salient objects and the initial one does not (See Figure 17 to visualize the different environments, and Figure 20 to visualize the groups of regions called areas in these explanations). We now display in Figure 20 the evolution of the proportion of time in each of these groups, for all of the experiments.

Regarding the curves obtained for both the **no objects** and **new objects** environments, the following behaviors can be observed.

- Between 0 and 200 observations, the exploration time drops in each area, except the one containing salient objects. This area is the most progressing one.
- After 900 observations, the area containing salient object decreases as well for the **'inf'** curve. This is because the model does not make much more progress in this area.

When looking in more details at the first row of the figure, obtained when switching to the **no objects** environment:

- For the curve **'100'**, after 200 observations, exploration decreases in the salient objects area and increases everywhere else. This is because this area does not contain any salient object anymore.
- After 200 observations, the complex background area now seems to be the most observed one.
- The same behavior is observed for curve **'400'**, after 600 observations.

Lastly, when examining the second row, obtained when switching to the **new objects** environment, the following comments can be made:

- For the curve **'100'**, after 200 observations, exploration decreases in the salient objects area and increases in the new salient objects area. This is because this area now contains much more salient objects and is now worth exploring more.
- The same behavior is observed for curve **'400'**, after 450 observations.

As a last general comment, we observe a certain delay (between 50 and 200 observations) between the actual switch of the environment, and the exploration behavior of the robot. This may be explained by the time required by the system to correctly assess the change in the learning progress. Therefore, the system is reactive to changes in the environment, but with a certain inertia.

D. Execution time

We now provide a table estimating the execution time for each module of the system. Our implementation is written in C++ and has been tested on Ubuntu 14.04 with an Intel Core i3-3240, CPU at 3.4GHz quadcore processor and with an Nvidia GTX Titan X graphic card. The online random forest has a training time that increases linearly with the amount of collected data. This part is therefore the bottleneck of the system. However, we consider in our architecture a different thread for training and for exploration, so that the robot can keep exploring while training is being performed.

VII. Conclusions

In this article, we have presented a full architecture for learning a model of visual saliency on a foveated platform called BioVision. The model of saliency is learned directly during the robot's exploration based on an object recognizer trained on the fovea. The signal provided by the recognizer is then transfered to the contextual view to produce saliency maps. Moreover, we investigated how the robot could methodically explore its environment to learn the saliency model faster and better. We proposed an original approach based on the IAC algorithm to guide exploration in that regard. We have carried out several experimentation to demonstrate the accuracy of our saliency maps as compared with other state-of-the-art approaches, and the efficiency of our exploration technique.

In a future work, we would like to investigate the possibility of using an end-to-end deep learning framework to build the model of saliency. We so far separate feature extraction and feature combination, but deep learning offers a way to integrate both at the same time. Additionally, neural networks are by essence online classifiers, which may be better-suited that the proposed method based on random forests that have been shown to be the current bottlenec in terms of speed computation. Second, we would like to carry out experiments in a non simulated setup to have a fully operational system. Lastly, we would like to integrate more biologically plausible saccadic models to pilot the gaze exploration of the robot. Indeed, saccades are biased by many factors that do not make the distribution of saccades homogeneous (for example horizontal and vertical saccades are much more common than oblique ones [59]).

Acknowledgment

The authors would like to thank the INRIA Flowers team, and especially Pierre-Yves Oudeyer for the valuable help on the IAC aspect. We also would like to thank Michael Fagno for making the BioVision platform operational.

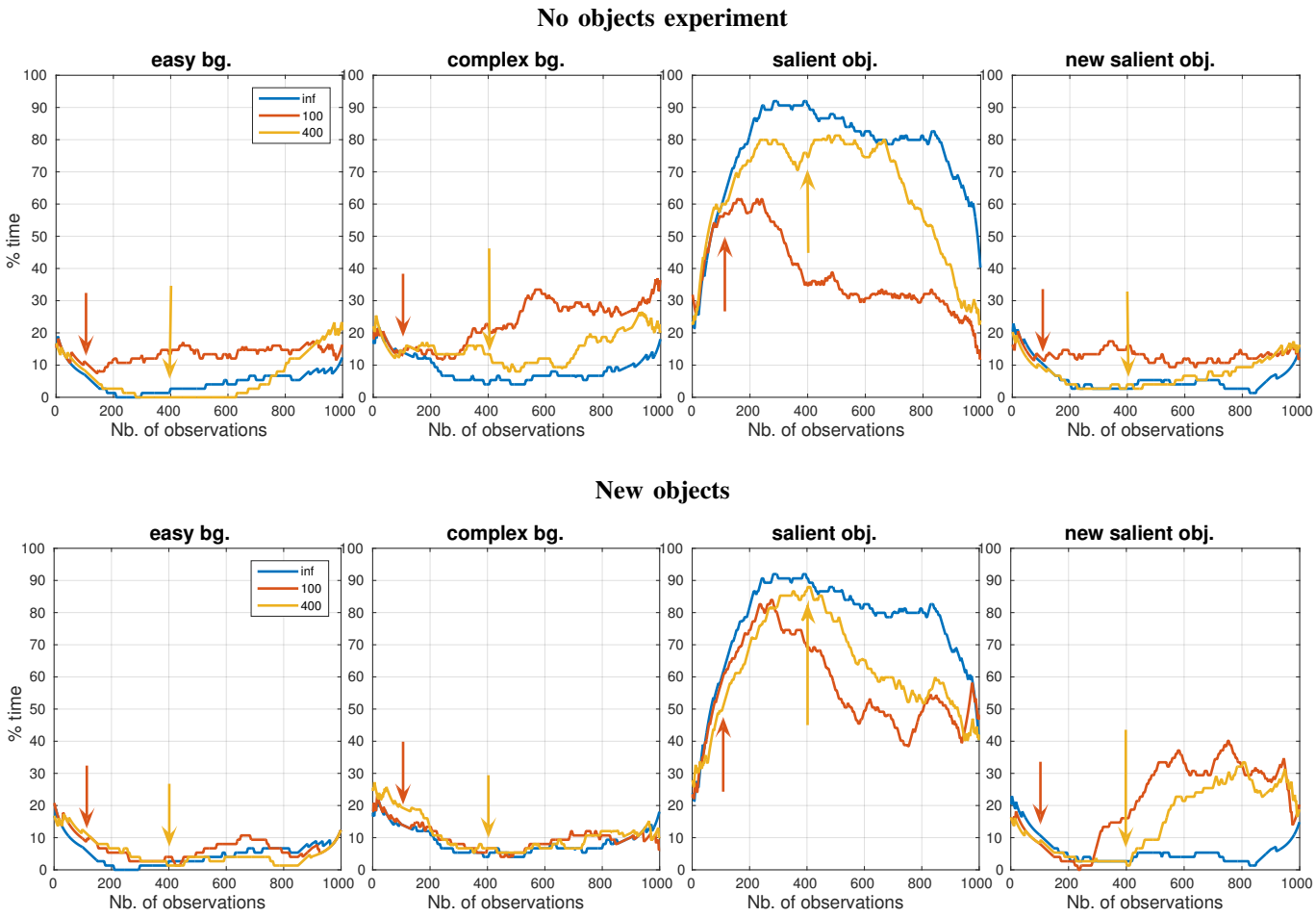

Fig. 20. Time spent in each group of regions for each experiment. Top: **no objects**. Bottom: **new objects experiments**. The red and yellow arrows point out the frame at which the change in the environment occurs.

Step	Min time	Max time	Comment
Feature extraction	26 ms	3 300 ms	Depends whether feature extraction is calculated on GPU or CPU
Foveal segmentation	12 ms	240 ms	Depends whether object recognition is calculated on GPU or CPU
Classifier update	23 ms	13 500 ms	Depends on the number of samples to re-train (between 1 and 100 000)
Saliency estimation	14 ms	82 ms	Depends on the features and if superpixel refinement is used(GPU only)

TABLE I
PROCESSING TIME OF THE MAIN STEPS OF THE SALIENCY LEARNING ALGORITHM

REFERENCES

[1] Adrien Baranès and P-Y Oudeyer. R-iac: Robust intrinsically motivated exploration and active learning. *Autonomous Mental Development, IEEE Transactions on*, 1(3):155–169, 2009.

[2] Adrien Baranes, Pierre-Yves Oudeyer, and Jacqueline Gottlieb. Eye movements reveal epistemic curiosity in human observers. *Vision research*, 117:81–90, 2015.

[3] M Bjorkman and Danica Kragic. Combination of foveal and peripheral vision for object recognition and pose estimation. In *Robotics and Automation, 2004. Proceedings. ICRA'04. 2004 IEEE International Conference on*, volume 5, pages 5135–5140. IEEE, 2004.

[4] Mårten Björkman and Jan-Olof Eklundh. Vision in the real world: Finding, attending and recognizing objects. *International Journal of Imaging Systems and Technology*, 16(5):189–208, 2006.

[5] Mårten Björkman and Danica Kragic. Active 3d scene segmentation and detection of unknown objects. In *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pages 3114–3120. IEEE, 2010.

[6] Aysecan Boduroglu, Priti Shah, and Richard E Nisbett. Cultural differences in allocation of attention in visual information processing. *Journal of Cross-Cultural Psychology*, 40(3):349–360, 2009.

[7] Ali Borji, Majid Nili Ahmadabadi, Babak Nadjar Araabi, and Mandana Hamidi. Online learning of task-driven object-based visual attention control. *Image and Vision Computing*, 28(7):1130–1145, 2010.

[8] Ali Borji and Laurent Itti. State-of-the-art in visual attention modeling. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 35(1):185–207, 2013.

[9] Zoya Bylinskii, Tilke Judd, Frédo Durand, Aude Oliva, and Antonio Torralba. Mit saliency benchmark. http://saliency.mit.edu/.

[10] José M Cañas, Marta Martínez de la Casa, and Teodoro González. An overt visual attention mechanism based on saliency dynamics. *International Journal of Intelligent Computing in Medical Sciences & Image Processing*, 2(2):93–100, 2008.

[11] Ming-Ming Cheng, Guo-Xin Zhang, Niloy J Mitra, Xiaolei Huang, and Shi-Min Hu. Global contrast based salient region detection. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 409–416. IEEE, 2011.

[12] Nuttapong Chentanez, Andrew G Barto, and Satinder P Singh. Intrinsically motivated reinforcement learning. In *Advances in neural information processing systems*, pages 1281–1288, 2004.

[13] Céline Craye, David Filliat, and Jean-François Goudou. Exploration strategies for incremental learning of object-based visual saliency. In

2015 Joint IEEE International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob), pages 13–18. IEEE, 2015.

[14] Celine Craye, David Filliat, and JF Goudou. Environment exploration for object-based visual saliency learning. In *Robotics and Automaton (ICRA), 2016 IEEE International Conference on*, pages 3140–3148, 2016.

[15] Celine Craye, David Filliat, and JF Goudou. Rl-iac: An exploration policy for online saliency learning on an autonomous mobile robot. In *Intelligent Robots and Systems (IROS), 2016 IEEE International Conference on*, 2016.

[16] Celine Craye. *Intrinsic motivation mecanisms for incremental learning of visual saliency*. PhD thesis, Paris Saclay, 2017.

[17] Erkut Erdem and Aykut Erdem. Visual saliency estimation by nonlinearly integrating features using region covariances. *Journal of vision*, 13(4):11, 2013.

[18] Simone Frintrop. *VOCUS: A visual attention system for object detection and goal-directed search*, volume 3899. Springer, 2006.

[19] Simone Frintrop, Erich Rome, and Henrik I Christensen. Computational visual attention systems and their cognitive foundations: A survey. *ACM Transactions on Applied Perception (TAP)*, 7(1):6, 2010.

[20] Simone Frintrop, Thomas Werner, and Germán Martín García. Traditional saliency reloaded: A good old model in new shape. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 82–90, 2015.

[21] Jacqueline Gottlieb, Pierre-Yves Oudeyer, Manuel Lopes, and Adrien Baranes. Information-seeking, curiosity, and attention: computational and neural mechanisms. *Trends in cognitive sciences*, 17(11):585–593, 2013.

[22] Stephen Gould, Joakim Arfvidsson, Adrian Kaehler, Benjamin Sapp, Marius Messner, Gary R Bradski, Paul Baumstarck, Sukwon Chung, and Andrew Y Ng. Peripheral-foveal vision for real-time object recognition and tracking in video. In *IJCAI*, volume 7, pages 2115–2121, 2007.

[23] Fred H Hamker. The emergence of attention by population-based inference and its role in distributed processing and cognitive control of vision. *Computer Vision and Image Understanding*, 100(1):64–106, 2005.

[24] MR Harter and L Anllo-Vento. Visual-spatial attention: preparation and selection in children and adults. *Electroencephalography and clinical neurophysiology. Supplement*, 42:183–194, 1990.

[25] Xiaodi Hou, Jonathan Harel, and Christof Koch. Image signature: Highlighting sparse salient regions. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 34(1):194–201, 2012.

[26] Xiao Huang and John Weng. Novelty and reinforcement learning in the value system of developmental robots. 2002.

[27] Nicolas Hueber, Pierre Raymond, Christophe Hennequin, Alexander Pichler, Maxime Perrot, Philippe Voisin, and Jean-Pierre Moeglin. Bio-inspired approach for intelligent unattended ground sensors. In *SPIE Sensing Technology+ Applications*, pages 94940S–94940S. International Society for Optics and Photonics, 2015.

[28] M Hulse, Sebastian McBride, James Law, and Mark Lee. Integration of active vision and reaching from a developmental robotics perspective. *Autonomous Mental Development, IEEE Transactions on*, 2(4):355–367, 2010.

[29] Laurent Itti and Christof Koch. Computational modelling of visual attention. *Nature reviews neuroscience*, 2(3):194–203, 2001.

[30] Laurent Itti, Christof Koch, and Ernst Niebur. A model of saliency-based visual attention for rapid scene analysis. *IEEE Transactions on pattern analysis and machine intelligence*, 20(11):1254–1259, 1998.

[31] Varan Kompella, Matthew Luciw, Marijn Stollenga, Leo Pape, and Jurgen Schmidhuber. Autonomous learning of abstractions using curiosity-driven modular incremental slow feature analysis. In *Development and Learning and Epigenetic Robotics (ICDL), 2012 IEEE International Conference on*, pages 1–8. IEEE, 2012.

[32] Varan R Kompella, Marijn F Stollenga, Matthew D Luciw, and Juergen Schmidhuber. Explore to see, learn to perceive, get the actions for free: Skillability. In *2014 International Joint Conference on Neural Networks (IJCNN)*, pages 2705–2712. IEEE, 2014.

[33] Danica Kragic. Object search and localization for an indoor mobile robot. *CIT. Journal of Computing and Information Technology*, 17(1):67–80, 2009.

[34] Danica Kragic and Marten Bjorkman. Strategies for object manipulation using foveal and peripheral vision. In *Computer Vision Systems, 2006 ICVS'06. IEEE International Conference on*, pages 50–50. IEEE, 2006.

[35] Danica Kragic, Mårten Björkman, Henrik I Christensen, and Jan-Olof Eklundh. Vision for robotic object manipulation in domestic settings. *Robotics and autonomous Systems*, 52(1):85–100, 2005.

[36] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[37] Srinivas SS Kruthiventi, Kumar Ayush, and R Venkatesh Babu. Deepfix: A fully convolutional neural network for predicting human eye fixations. *arXiv preprint arXiv:1510.02927*, 2015.

[38] Matthias Kümmerer, Lucas Theis, and Matthias Bethge. Deep gaze i: Boosting saliency prediction with feature maps trained on imagenet. *arXiv preprint arXiv:1411.1045*, 2014.

[39] Kevin Lai, Liefeng Bo, Xiaofeng Ren, and Dieter Fox. A large-scale hierarchical multi-view rgb-d object dataset. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 1817–1824. IEEE, 2011.

[40] Balaji Lakshminarayanan, Daniel M Roy, and Yee Whye Teh. Mondrian forests: Efficient online random forests. In *Advances in Neural Information Processing Systems*, pages 3140–3148, 2014.

[41] Mikko Lauri and Risto Ritala. Stochastic control for maximizing mutual information in active sensing. In *IEEE Int. Conf. on Robotics and Automation (ICRA) Workshop on Robots in Homes and Industry*, 2014.

[42] Manuel Lopes, Tobias Lang, Marc Toussaint, and Pierre-Yves Oudeyer. Exploration in model-based reinforcement learning by empirically estimating learning progress. In *Advances in Neural Information Processing Systems*, pages 206–214, 2012.

[43] Nikolaos A Massios, Robert B Fisher, et al. *A best next view selection algorithm incorporating a quality criterion*. Department of Artificial Intelligence, University of Edinburgh, 1998.

[44] Silviu Minut and Sridhar Mahadevan. A reinforcement learning model of selective visual attention. In *Proceedings of the fifth international conference on Autonomous agents*, pages 457–464. ACM, 2001.

[45] Sara Mitri, Simone Frintrop, Kai Pervolz, Hartmut Surmann, and Andreas Nuchter. Robust object detection at regions of interest with an application in ball recognition. In *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, pages 125–130. IEEE, 2005.

[46] Sao Mai Nguyen, Serena Ivaldi, Natalia Lyubova, Alain Droniou, Damien Gerardeaux-Viret, David Filliat, Vincent Padois, Olivier Sigaud, and Pierre-Yves Oudeyer. Learning to recognize objects through curiosity-driven manipulation with the icub humanoid robot. In *Development and Learning and Epigenetic Robotics (ICDL), 2013 IEEE Third Joint International Conference on*, pages 1–8. IEEE, 2013.

[47] Maxime Oquab, Léon Bottou, Ivan Laptev, and Josef Sivic. Is object localization for free?-weakly-supervised learning with convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 685–694, 2015.

[48] P-Y Oudeyer, Frédéric Kaplan, and Verena Vanessa Hafner. Intrinsic motivation systems for autonomous mental development. *Evolutionary Computation, IEEE Transactions on*, 11(2):265–286, 2007.

[49] Lucas Paletta, Gerald Fritz, and Christin Seifert. Q-learning of sequential attention for visual object recognition from informative local descriptors. In *Proceedings of the 22nd international conference on Machine learning*, pages 649–656. ACM, 2005.

[50] Babak Rasolzadeh, Mårten Björkman, Kai Huebner, and Danica Kragic. An active vision system for detecting, fixating and manipulating objects in the real world. *The International Journal of Robotics Research*, 29(2-3):133–154, 2010.

[51] Nicolas Riche, Matthieu Duvinage, Matei Mancas, Bernard Gosselin, and Thierry Dutoit. Saliency and human fixations: state-of-the-art and study of comparison metrics. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1153–1160, 2013.

[52] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *arXiv preprint arXiv:1409.0575*, 2014.

[53] Amir Saffari, Christian Leistner, Jakob Santner, Martin Godec, and Horst Bischof. On-line random forests. In *Computer Vision Workshops (ICCV Workshops), 2009 IEEE 12th International Conference on*, pages 1393–1400. IEEE, 2009.

[54] Uta Sailer, J Randall Flanagan, and Roland S Johansson. Eye–hand coordination during learning of a novel visuomotor task. *The Journal of neuroscience*, 25(39):8833–8842, 2005.

[55] Jürgen Schmidhuber. Curious model-building control systems. In *Neural Networks, 1991. 1991 IEEE International Joint Conference on*, pages 1458–1463. IEEE, 1991.

[56] Jürgen Schmidhuber. Formal theory of creativity, fun, and intrinsic motivation (1990–2010). *Autonomous Mental Development, IEEE Transactions on*, 2(3):230–247, 2010.

[57] Michael Van den Bergh, Xavier Boix, Gemma Roig, Benjamin de Capitani, and Luc Van Gool. Seeds: Superpixels extracted via energy-driven sampling. In *Computer Vision–ECCV 2012*, pages 13–26. Springer, 2012.

[58] Sethu Vijayakumar, Jörg Conradt, Tomohiro Shibata, and Stefan Schaal. Overt visual attention for a humanoid robot. In *Intelligent Robots and Systems, 2001. Proceedings. 2001 IEEE/RSJ International Conference on*, volume 4, pages 2332–2337. IEEE, 2001.

[59] Wei Wang, Cheng Chen, Yizhou Wang, Tingting Jiang, Fang Fang, and Yuan Yao. Simulating human saccadic scanpaths on natural images. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 441–448. IEEE, 2011.

[60] Chong Zhang, Yu Zhao, Jochen Triesch, and Bertram E Shi. Intrinsically motivated learning of visual motion perception and smooth pursuit. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1902–1908. IEEE, 2014.

[61] Jianming Zhang and Stan Sclaroff. Saliency detection: a boolean map approach. In *Computer Vision (ICCV), 2013 IEEE International Conference on*, pages 153–160. IEEE, 2013.

[62] Qi Zhao and Christof Koch. Learning a saliency map using fixated locations in natural scenes. *Journal of vision*, 11(3):9, 2011.

[63] B. Zhou, A. Khosla, Lapedriza. A., A. Oliva, and A. Torralba. Learning Deep Features for Discriminative Localization. *CVPR*, 2016.

[64] Jun-Yan Zhu, Jiajun Wu, Yichen Wei, Eric Chang, and Zhuowen Tu. Unsupervised object class discovery via saliency-guided multiple class learning. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 3218–3225. IEEE, 2012.

Celine Craye Céline Craye received the diplome dIngénieur in telecommunications from the Ecole Nationale Supérieure des Télécommunications de Bretagne, France, in 2012. She received her M.A.Sc. degree from the Electrical and Computer Engineering Department at the University of Waterloo in 2013. She is currently a Ph.D. candidate at ENSTA Paristech in France at the UIIS laboratory. The Ph.D. is granted by the CIFRE program in partnership with Thales Vision Lab. Her research interests are in computer vision, machine learning and developmental robotics.

David Filliat David Filliat graduated from the Ecole Polytechnique in 1997 and obtained a PhD on bio-inspired robotics navigation from Paris VI university in 2001. After 4 years as an expert for the robotic programs in the French armament procurement agency, he is now professor at Ecole Nationale Suprieure de Techniques Avances ParisTech. Head of the Robotics and Computer Vision team since 2006, he obtained the Habilitation Diriger des Recherches en 2011. He is also a member of the ENSTA ParisTech INRIA FLOWERS team. His main research interest are perception, navigation and learning in the frame of the developmental approach for autonomous mobile robotics. http://www.ensta-paristech.fr/~filliat/

Jean-Francois Goudou Jean-Francois Goudou was born in Versailles, France, in 1979. He received the M.E. degree in applied mathematics from the Ecole Polytechnique, Palaiseau, France, in 2004, and the Ph.D. degrees in computer vision from Telecom ParisTech, Paris, France, in 2007. In 2008, he joined the Advanced studies department Theresis, from Thales, as a project manager for collaborative national and European projects. He is since 2009 in charge of the demonstrators of the VisionLab, the Vision innovation laboratory for video-surveillance of the Thales Group. He is also leading several research projects in the topics of video-surveillance and algorithmic evaluation. He is since 2016 deputy head of the VisionLab, in charge of academic co-operations and H2020 projects proposals and management. His research interest include the complete sensing chain from camera to processing, the human vision and bio-inspired processing and neural networks, including deep learning.