CNN-based novelty detection for terrestrial and extraterrestrial autonomous exploration

Bampis, Loukas; Gasteratos, Antonios; Boukas, Evangelos

Published in:
IET Cyber-systems and Robotics

Link to article, DOI:
10.1049/csy2.12013

Publication date:
2021

Document Version
Publisher's PDF, also known as Version of record

Link back to DTU Orbit

Citation (APA):

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.
CNN-based novelty detection for terrestrial and extra-terrestrial autonomous exploration

Loukas Bampis\textsuperscript{1} \hspace{1em} Antonios Gasteratos\textsuperscript{1} \hspace{1em} Evangelos Boukas\textsuperscript{2}

\textsuperscript{1}Department of Production and Management Engineering, Democritus University of Thrace, Xanthi, Greece
\textsuperscript{2}Department of Electrical Engineering, Technical University of Denmark, Kongens Lyngby, Denmark

Correspondence
Loukas Bampis, 12 Vas. Sophias, GR-671 32, Xanthi, Greece.
Email: lbampis@pme.duth.gr

Funding information
"Novelty or Anomaly Hunter" (NOAH)

Abstract
Novelty detection is concerned with detecting features that do not belong to any known class or are not well represented by existing models. Ergo, in autonomous navigation novelty detection determines whether an input camera frame contains certain entities of high interest which do not correspond to a known category. One of the key requirements for the future space exploration missions is the reduction of the information to be transferred back to Earth. Thus, novelty detection techniques have been developed to select the subset of acquired images with significant measurements that justify utilisation of the limited bandwidth from the available information link. Such methods are based on the identification of salient regions, which are then evaluated against a set of trained classifiers. We explore a novelty detection approach, based on the reasoning properties of Neural Networks, which follow the same guidelines while also being trainable in an end-to-end manner. This characteristic allows for the intertwined optimisation of the individual components leading to a closer estimation of a global solution. Our experiments reveal that the proposed novelty detection system achieves better performance, as compared to hand-crafted techniques, when the learning and testing examples refer to similar environments.

1 | INTRODUCTION

Planetary Exploration Missions require advanced perception to overcome distance-induced latency and bandwidth limitations. Science missions, such as NASA's Mars Exploration Rovers (MER) and Mars Science Laboratory (MSL), as well as the upcoming ESA's (European Space Agency) Exomars, search for biologically and geologically interesting areas, which may provide new knowledge about alien life and planet formation. Planetary rovers should (a) autonomously navigate within their environment \cite{1} and (b) autonomously detect regions of interest for further examination \cite{2}. Hitherto, active missions' attempts to perform autonomous detection have been successful in identifying areas being either specifically hard-coded in or trained upon before the mission \cite{3,4}. However, in planetary robotic exploration—as is the case with any exploration attempt—the most significant findings are the unexpected ones. So far, such detections have been mainly performed by human experts. This is due to bandwidth limitations hindering the full downstream of data stemming from the rovers' perception systems for offline analysis. To that end, the Novelty or Anomaly Hunter (NOAH) activity, run by the ESA, aims to provide a system that can automate the accurate selection of those information-rich data. There have been previous attempts for novelty detection, which included either various levels of tuneable parameters or heuristic approaches. In the work described here, we implement the first CNN-based parameter-free approach to address the problem of novelty detection.

The latter constitutes a challenging task in autonomous systems and still remains an open issue in machine vision. The related literature on this matter is mostly limited to techniques for identifying if a certain measurement should be characterised as a novel one based on its resemblance to training data \cite{5} or an image from the input camera stream is sufficiently dissimilar (as a whole) from the rest \cite{6}. One of the few architectures that tackles the novelty detection task on a robot has been proposed during the Mobile Autonomous Scientist for Terrestrial and Extra-terrestrial Research (MASTER) activity...
Its scope was not purely restricted to the annotation of an object as novel or not, but set a complete pipeline for detecting entities of interest, as well as measuring their resemblance against known data. More specifically, the pipeline proposed within MASTER envisaged novelty detection as a three-stage procedure (Figure 1b). Given an input image, saliency detection mechanisms [8–10] were developed to identify and segment Regions of Interest (RoIs) which formulate object proposals. Then, a set of trained Support Vector Machine (SVM) classifiers is applied measuring the membership of each RoI in the known categories. Last, novelty rules were applied to the classification output, assigning labels to the identified entities, spanning from “Expected” (known object) to “Unclassified Novelty” (unknown object). The first two of the above modules perform in discrete and subsequent steps, which need to be individually fine-tuned so as to achieve high performance.

In this study, we extend the aforementioned processing steps and present an alternative approach, which can be used to simultaneously detect RoIs (even if they correspond to unknown objects) and measure their membership to a set of predefined classes (Figure 1b). Thus, we make use of the Faster R-CNN architecture [11]; an end-to-end trainable model based on the inferencing properties of Neural Networks (NNs). Like any other cost–function minimisation scheme, Faster R-CNN can offer improved system performance, as compared to handcrafted methods, due to the following characteristics:

- The parameterisation for every different component of the architecture is simultaneously tuned to advance the final system’s performance (end-to-end training).
- The cost-function to be minimised can estimate the specific functionality desired by the system.
- Changing the targeted operational environment does not impose altering the algorithm's complexity. Given the appropriate training data, the system can adapt.

In Section 2, the evaluated technique for detecting novelties is briefly reviewed, while Section 3 describes the process for integrating such an architecture within the NOAH pipeline. The evaluation of the proposed approach is presented in Section 4, and finally, Section 5 gives the conclusions.

## 2 Faster R-CNN Architecture

The goal of Faster R-CNN, as initially proposed by the respective authors, was originally to develop an NN to efficiently detect and classify objects in images. The classification was performed with a set of predefined classes regardless of the subjects’ positioning inside the frame. For a complete description of Faster R-CNN one can refer to the respective paper [11], yet for completeness, the authors discuss here the technique’s critical components and highlight the characteristics that render the architecture suitable for NOAH activity. As shown in Figure 2a, Faster R-CNN can essentially be segmented into two distinct modules, namely, a Region Proposal Network (RPN) and a Classification Network (CN). The RPN is responsible for identifying bounding-boxes that contain objects, while the CN operates as a classifier on the regions proposed by RPN. While both these networks could be completely isolated from one another, they share convolutional layers with the aim to gain in execution time without comprising their respective performances.

### 2.1 Region Proposal Network

Considering the case of a VGG16 model [12] as an example case for our description, Figure 2b shows the architecture of the RPN network. As a preprocessing step, each input image is rescaled such that the shorter dimension becomes \(d = 600\) pixels, while retaining the original aspect ratio. Using only the first convolutional layers of the VGG16 network, the architecture produces a feature map \(F\). Then, a \(3 \times 3\) convolutional sliding window is applied over \(F\) resulting in a set of relative locations a bounding-box can be potentially detected. Instead of predicting a single region proposal per sliding window location, Faster R-CNN introduces the mechanism of *anchors*. Anchors are defined as \(k = 3 \times 3 = 9\) reference boxes per sliding window instance corresponding to three different scales and three different aspect ratios. The resulting number of considered anchors is significantly reduced to 512 by retaining only 256 positive and 256 negative examples. Note that an anchor is characterised as positive or negative based on the area it shares with the Ground-Truth (GT) annotations. Last, this dimensionally reduced 512-days intermediate layer is followed by a ReLU [13] layer and two fully connected ones, producing the final output. Those values define the proposed bounding-boxes, together with an additional “objectness” measurement \((O \in (0, 1))\), which quantifies the existence of an object within each one of them. Finally, in order to reduce RPN proposals with highly overlapping regions, Non-Maximum Suppression (NMS) is adopted. In particular, an

![FIGURE 1](image)
Intersection over Union (IoU) threshold $r$ for NMS is used, and the remaining RoIs are propagated to the CN for classification.

2.2 Classification Network (CN)

Faster R-CNN utilises the Fast R-CNN architecture [14] to classify the detected regions proposed by the RPN. The bounding-boxes are used to extrapolate the corresponding RoIs onto the computed $F$ for a particular image. Thus, a set of $S$ feature maps is generated through the projection of the RPN proposals on $F$. Fast R-CNN uses an RoI pooling layer (Figure 2c) to reduce the varying-size members of $S$ into fixed-size ($7 \times 7$) feature sub-maps using max-pooling. Each sub-map is then fed into a sequence of fully connected layers which finally lead into two sibling output layers. The first one produces softmax probabilities over a predefined set of $K$ classes, while the second one provides an additional regression to refine the input bounding-box's position.

2.3 Network training

Training the Faster R-CNN architecture requires two cost functions, each of which is responsible for one of the RPN and CN modules.

2.3.1 RPN module

Training the RPN model requires the definition of positive and negative learning anchor examples. To that end, each used anchor is assigned with a binary label (object or non-object) based on two rules. Positive anchors are defined as the ones that: (i) have the highest IoU overlap or (ii) result in IoU > 0.7 with any ground-truth bounding-box. On the contrary, negative anchors are the ones with IoU < 0.3 against all ground-truth boxes. The rest does not contribute to the training procedure. The cost function minimisation scheme proposed in [11] is followed, which is governed by Stochastic Gradient Descent (SGD) [15]:

$$L(p_i; t_i) = \frac{1}{N_{cls}} \sum_i L_{cls}(p_i; p_i^*) + \lambda \frac{1}{N_{reg}} \sum_i p_i^* L_{reg}(t_i; t_i^*)$$

In the above equation, the first summation term corresponds to the classification part of the RPN model. Each considered anchor's index in a mini-batch is denoted as $i$, with $p_i$ being its predicted objectness probability. The ground-truth objectness $p_i^*$ is assigned with 1 for the positive examples and 0 for the negative ones, while the classification loss $L_{cls}$ is a log-loss over the two possible classes (object or non-object), with $L_{cls} = -\log(p_i)$ for a positive example. Note that the normalisation term $N_{cls}$ is fixed to the selected mini-batch size.

Regarding the regression part of the model (second summation term), the $t_i$ vector denotes the four coordinate parameters of the predicted bounding-box, and $t_i^*$ the ones of the ground-truth box that the corresponding anchor is associated with. The $L_{reg}$ loss is the smooth $L1$ function proposed in [14]. Note that the role of $p_i^* L_{reg}$ multiplication is to eliminate the regression loss in cases of negative examples. For positive samples, the loss is additionally normalised by $N_{reg}$, which equals the number of anchor locations and is weighted by $\lambda$ to impact the total cost function in the same degree with $L_{cls}$.
2.3.2 | CN module

As already mentioned, the CN part of Faster R-CNN is constituted by a Fast R-CNN architecture with the following cost function applied on every propagated bounding-box:

\[ L(r, r^*, t_r, t^*_r) = L_{cls}(r, r^*) + \lambda^r \left[ r^* \geq 1 \right] L_{reg}(t_r, t^*_r) \]

(2)

By analogy, the first summation term refers to a classification layer of \( K + 1 \) classes and \( L_{cls} \) denotes a log-loss. Variables \( r \) and \( r^* \) are the predicted probabilities for each class and the corresponding ground-truth label, respectively. The second summation term offers an additional per-class regression loss \( L_{reg} \) to the predicted bounding-boxes of RPN. Note that \( t_r \) and \( t^*_r \) denote a scale invariant translation and a log-space width/height shift relative to a proposed object. The term \( \left[ r^* \geq 1 \right] \) evaluates to 1 for \( r^* \geq 1 \) and 0 otherwise, in order to suppress the regression output in cases of background proposals (negative examples), where \( r^* = 0 \). Finally, the \( \lambda^r \) parameter is used to balance the two aforementioned losses.

2.3.3 | Training with shareable features

An essential characteristic of the method is its capability to share convolutional layers between these two networks. To that end, two main approaches were developed, so as to allow the convergence of shareable features into a network that favours both architectures. On the one hand, the alternating training (alt) learns the weights by interchanging the cost functions and their output on a sequential manner. On the other hand, the approximate\(^1\) joint training (end-to-end) relies on the idea of merging RPN and CN into a single network, combining their loss signals once the convolutional layers are reached.

3 | FASTER R-CNN WITHIN NOAH

In this section, the benefits of adopting a trainable method, such as Faster R-CNN, for NOAH are discussed. As stated before, a typical novelty detection algorithm is structured in three processing steps. First, a saliency detection mechanism identifies RoIs that deviate from the rest of the input image's content. Then, a set of binary classifiers, trained on recognising known entities with no particular scientific interest, is applied over each of the RoIs. Finally, novelty detection rules are applied, characterising a salient region as an abnormality if the classification results are not conclusive. The outcome of these two sub-networks can straightforwardly substitute the first two mechanisms of a novelty detection pipeline, offering a unified trainable system.

---

\(^1\)The end-to-end approach assumes zero back-propagation derivatives of the RoI pooling layer, which was found sufficient due to the system's direct supervision.

3.1 | RPN as a saliency detector

In order to identify an RoI within an image the objectness measurement produced by RPN can be used as a training method, instead of measuring the saliency of a particular region. On this account, we highlight the main differences between the mechanisms of “objectness” and “saliency”:

- **Objectness is adaptive:** For an entity to be detected through saliency, the respective region needs to obey a set of handcrafted rules. As shown in the results presented in many saliency detection techniques, a single saliency detection method fails to offer a global solution since it is static and predefined \([8–10,16–18]\). This essentially means that in order to advance the system's detection performance, the adopted saliency algorithm needs to be altered or completely substituted. On the contrary, a trainable measurement like the objectness, since derived from an Neural Network (NN), can potentially estimate any desired property given the appropriate learning examples.

- **Objectness is not driven by the images' overall content:** Many saliency detection approaches (e.g. \([17,18]\)) compare intensity and texture measurements between different image regions to highlight an RoI. This implies that the detection accuracy is relative to the scene's overall content. Besides, objectness is an absolute value remaining uninfluenced by the existence or absence of other objects.

- **Objectness can distinguish overlapping entities:** Due to the design of *anchors*, the RPN proposals can distinguish between objects sharing regions on an image. The equivalent attribute is not straightforward when considering a saliency detection technique, in which the augmentation of salient areas in bounding-boxes can merge multiple objects per detection and distort the results \([19]\). This characteristic is of great importance for a system like NOAH, where abnormalities can be located on top of known objects (e.g., a texture irregularity on a rock).

Despite the advantages listed before, there exists a discontinuity between the original objective of RPN architecture and the goal of NOAH. RPN was originally intended to identify objects already learnt during the networks training procedure, while NOAH seeks abnormalities (i.e., entities that may have never been encountered before). Thus, a critical question arises: “Can objectness highlight objects that the network has never been trained on?" To answer that, a series of experiments are presented in Section 4, yet we need to underline the information objectness is referring to. During training, the RPN loss function accepts bounding-boxes from any class of the GT as positive examples. Therefore, objectness is class-agnostic and only learns to seek those attributes that distinguish an object from the background. Those attributes are restricted to be shared among every class of the training set, leading to the global characteristics held by any object within a given environment.
3.2 CN as a set of binary classifiers

MASTER pipeline applied a set of binary classifiers on each detected RoI. Faster R-CNN incorporates the same intrinsic functionality with the classification output of CN layers. Combining RPN and CN into a single structure allows for the latter network to be effectively tuned with respect to the output of the first one. In addition, the regression CN output can also be used to better localise the detections. While promising, CNN performance is heavily dependent on a carefully formulated training procedure. All CNN-based classifiers require a vast amount of learning data and can be biased towards the class taking in the most GT examples.

4 RESULTS

In this section, the performance of Faster R-CNN is tested. We focus our experimentation on the evaluation of RPN capability to extend its proposals on unknown objects, that is, not included in the learning examples. As mentioned, this is the central aspect to justify the method's applicability on novelty detection. To assess a particular approach, the detection performance was measured using the area under the Receiver Operating Characteristic (ROC) curve (AUC), due to its high tolerance over a class's skew [20]. In the majority of our experiments, we measured a per-pixel system performance. Given a testing image, True-Positive (TP) detections correspond to pixels that simultaneously belong to an annotated GT region and assigned with a score higher than a predefined threshold. On the contrary, False-Positive (FP) pixels are characterised as proposals with the same objectness, though, not belonging to any GT region. Finally, given a specific set of system parameters, a single point in the ROC curve can be computed by averaging the achieved TP and FP detections from every testing image. Note that for a detection system to be meaningful, AUC values should be > 0.5 (50%).

We begin by discussing the selected training data in order to support the above-described evaluation. Subsequently, RPN's performance is presented and compared with another well-established saliency detection approach to highlight its competitiveness over the typical pipeline for finding novelties [7]. To further assess its detection capabilities, Faster R-CNN is also evaluated on the identification of overlapping entities. Despite the unavailability of sufficient learning examples, we additionally deploy the network on data sets referring to Mars and Mars-analogue environments, exceeding earlier approaches' performance. Finally, we present an extensive set of experiments to quantify the effect of different hyperparameters over novelty detection effectiveness while also proving their negligible impact on classification performance.

4.1 Training data

Any CNN-based architecture requires a well-defined training sample in order to perform well during online operation. The selected learning examples need to be consistently labelled, contain enough information for each included class and comply with the testing environment's properties. At the time of writing, such a set of images is not available for the environment of Mars. Thus, until activities like LabelMars [21] are concluded, we wish to assess the capability of Faster R-CNN on detecting unlearned, though meaningful, objects through RPN using alternative evaluation environments. For this reason, the PASCAL Visual Object Classes 2007 (VOC2007) benchmark [22] is used. This data set was produced within the scope of the namesake competition and offers great labelling characteristics. The contained ~10,000 images are in their majority RGB, though a few monochromatic instances can also be found. In total, 20 classes are included with every instance being annotated by a GT bounding-box.

4.2 Measuring the detection accuracy of unknown objects

4.2.1 Evaluation procedure

To assess our hypothesis of an RPN being capable of detecting unknown objects, we should not utilise the standard training and testing pipeline of Faster R-CNN. Instead of learning on a subset of the available examples from a particular class and then measuring the detection accuracy on the remaining instances, we need to render the network oblivious to specific kinds of objects. For this reason, given a data set with C classes annotated in GT, our testing pipeline was formulated as follows:

1. Separate a class c from C. The remaining set of classes is denoted as C−.
2. Train the Faster R-CNN architecture using C−.

| TABLE 1 Properties and key differences of the two evaluated VOC2007 classes |
|-----------------|-----------------|
| **Class: Cow** | **Class: Train** |
| Typically enclosed in uncluttered environments | Typically enclosed in cluttered environments |
| The object of interest (cow) is not composed from multiple sub-objects | The object of interest (train) is composed from multiple sub-objects (windows, chimneys, headlights, etc.) |
| Typically include multiple distant class instances per frame | Typically include a single class instance per frame |
3. Evaluate the RPN capability to detect instances of the unlearned class $c$.

This procedure simulates the detection of an unknown (novel) object. Note that all the shareable convolutional layers were initialised from a Zeiler and Fergus [23] (ZF) network trained on the ImageNet classification data set [24], while the rest of the neurons were sampled via a Gaussian distribution $G_R(\mu_R = 0, \sigma_R = 0.01)$. In our experiments, an additional NMS was applied to the final detections using an IoU threshold $r = 0.3$ (suppressing boxes with IoU $\geq 0.3$) since bounding boxes with higher overlap are typically originated by the same physical entity from the observed environment. Note that this is also in accordance with the setup described within the training procedure (Section 2.3.1), where anchors were considered positive if they shared an IoU $\geq 0.3$ with a ground-truth object. The rest of the parameters were pulled from the original implementation [11]. In order to test our assumption on classes with different properties, our evaluation was executed twice, each time excluding a different class of the VOC2007 data set. The corresponding $c$ class for each iteration was either the one of “cow” or “train’, with Table 1 listing their key differences and most illustrative attributes. Figure 3 contains representative examples from each case. As stated, within VOC2007, each instance of the aforementioned classes is guaranteed to be annotated in every image. Nevertheless, there exist some apparent objects (e.g. buildings or street signs) not belonging to the selected 20 class labels and, thus, not included in the GT. For this reason, some RPN proposals, although meaningful, are predestined to be characterised as FP detections reducing the performance metrics.

4.2.2 | Performance

The ROC curves in Figure 4 resulted from varying an objectness threshold on the RPN proposals. Note that in all ROC figures, red lines indicate minimum meaningful performance. As can be seen, the achieved detection accuracy of the “train” class is lower than the “cow” one due to the highly cluttered environment captured in the background. Additionally, we evaluated a representative saliency detection method [18] on the same two classes, with a view to provide a baseline performance for the same testing set. By inspecting the ROC curves in Figure 5, it is apparent that the RPN proposals perform better than a state-of-the-art saliency detection method, even though the network has never encountered any learning examples from the corresponding objects of interest. The same experimental procedure (training on $C$ and testing on $c$) was repeated over the VOC2007 images after converting them to greyscale. This series of experiments was conducted to further simulate the operational conditions of NOAH since the targeted camera sensor is expected to be monochromatic. The achieved performance is slightly reduced (Figure 6) implying that colour information, while helpful, is not a decisive attribute learnt by the system.

In Section 3.1, we mentioned the RPN beneficial property of detecting overlapping objects (Figure 7). For two overlapping detections to be meaningful, the produced bounding-
boxes need to enclose the respective objects of interest in a sufficient percentage, allowing the classifier to determine the RoI’s membership to one of the predefined classes (if not novel). Positive class examples during the CN training are considered the proposals achieving IoU > 0.5 with a GT annotation [14]. That being said, the classifier can identify even smaller subregions of an object during testing, though we are still interested in a more rigorous evaluation. Since, for this case, the total number of possible detections cannot be precisely computed, we made use of Precision-Recall (PR) curves (Figure 8) and report the achieved mean Average Precision (AP) metrics. More specifically, for this experiment, we accepted as TP the detections sharing a sufficient number of pixels with a GT annotation (IoU > 0.5 and IoU > 0.7), while the rest were treated as FP ones. For a given objectness threshold, the achieved TP and FP rates were averaged among every available image in the data set. As the results indicate, the RPN architecture can distinguish between objects sharing certain image regions. The achieved detection performance in the case of “train” is reduced since the respective class comprises of multiple non-labelled sub-objects (e.g. chimneys and headlights), as shown in Figure 7b.

As a final note, we extend our evaluation procedure presented in Section 4.2.1 with the view to measure the proposed system’s response over the detection of multiple unknown

---

**FIGURE 5** Receiver operating characteristic curves measuring the saliency detection performance of method [18] on the two testing classes. (a) Testing class: “cow” (b) Testing class: “train”

**FIGURE 6** Receiver operating characteristic curves measuring the detection performance of region proposal network on greyscale images that contain unknown classes. (a) Unknown class: “cow” (b) Unknown class: “train”

**FIGURE 7** Samples of overlapping objects in VOC2007-evaluated classes. Ground-truth (GT) annotations and region proposal network proposals are marked. (a) Testing class: “cow” (b) Testing class: “train”
Towards this end, we removed the instances of both “cow” and “train” classes from the full set of VOC2007, and we trained Faster R-CNN on the remaining samples. The detection results from these two classes' entities are shown in Figure 9 and reveal that the network was still capable of preserving its novelty discovery capabilities. Note that a small performance reduction is to be expected since, in this case, the total amount of training samples is lower than the rest of the experiments.

**4.3 | VOC2007 applicability on Mars-analogue environments**

Albeit the respective environments are completely different, we stretched the limits of our experimentation looking for an immediate performance gain for NOAH activity. Thus, we trained Faster R-CNN on VOC2007 and evaluated the detection capabilities of RPN on three available Mars-analogue data sets, namely rover images of Planetary Science (PS) [25], SAFER (PSSafer) [26] and SEEKER (SK) [27]. Some representative instances for each of the above sets are illustrated in Figure 10. Table 2 shows the average AUC for each data set after the computation of the respective ROC curves. Although the learning examples correspond to a completely different
environment (e.g. different lighting, materials, texture, shape), the network was still able to identify some of the included annotations. In fact, since the above data sets contain some GT labels not corresponding to objects, a fair evaluation was performed by removing non-object label entries, namely: “sky”, “surface” and “drift”. It should be noted that RPN is capable of detecting those entries as well, if similar examples were included during the training procedure. Lastly, Table 2 contains comparative results with some of the most representative saliency measuring methods in the literature [8,17,18], which have been used for novelty detection. This transfer-learning scheme does not perform as competitively as the typical approach of training and testing on data sets with common attributes. Yet, the results are still promising, given that the network has never encountered similar scenes, thus proving the system’s capacity to detect unlearned objects even in the case of a completely unknown environment.

<table>
<thead>
<tr>
<th>Approach</th>
<th>Planetary science (PS)</th>
<th>SAFER PS (PSSafer)</th>
<th>SEEKER PS (SK)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Faster R-CNN [11] (full data sets)</td>
<td>0.535</td>
<td>0.566</td>
<td>0.574</td>
</tr>
<tr>
<td>Faster R-CNN [11] (excluding non-objects)</td>
<td>0.585</td>
<td>0.724</td>
<td>0.592</td>
</tr>
<tr>
<td>Schauerte et al. [8] (Multiscale Quaternion)</td>
<td>0.590</td>
<td>0.420</td>
<td>0.610</td>
</tr>
<tr>
<td>Schauerte et al. [8] (Quaternion)</td>
<td>0.560</td>
<td>0.470</td>
<td>0.620</td>
</tr>
<tr>
<td>Yang et al. [17]</td>
<td>0.567</td>
<td>0.531</td>
<td>0.593</td>
</tr>
<tr>
<td>Cheng et al. [18] Hist. Contrast</td>
<td>0.561</td>
<td>0.607</td>
<td>0.537</td>
</tr>
<tr>
<td>Cheng et al. [18] region contrast</td>
<td>0.576</td>
<td>0.581</td>
<td>0.573</td>
</tr>
</tbody>
</table>

4.4 Parameterisation

The evaluation results presented in Ren et al. [11] concluded to a specific set of best-performing parameters for the Faster R-CNN training and testing architecture. However, there is no guarantee that the same parameterisation is optimum for NOAH, since the desired functionality is the detection of unknown objects. Thus, we assessed the effect of four main aspects to the final performance of the pipeline, namely: (i) the depth of the network’s architecture (VGG16 as a deep configuration and ZF as a shallow one), (ii) the effect of different NMS thresholds during testing, (iii) the selected training approach and (iv) the learning examples’ scaling factors s that are individually applied on the input images. Since the effect of each parameter on the final accuracy was unknown beforehand, every possible combination was separately measured in a brute-force manner to identify the most promising setup. During this experiment, the achieved performance was assessed by excluding either the “cow” or the “train” class from training. In order to further simulate the operational conditions targeted by NOAH, the experiments were conducted after converting all images to greyscale.

Figure 11a portrays the average AUC metrics when no additional scaling of the training examples was introduced. Using a deep convolutional network such as VGG16 with alt training yields beneficial overall performance for both novel classes. Furthermore, an NMS threshold of 0.2 provides the best trade-off between excluding overlapping regions and retaining the most representative proposal boxes (AUCs of 0.84,208 for “cow” and 0.77,529 for “train”). When the available training examples are scaled over multiple factors s = {400, 500, 600, 700} (Figure 11b), an additional performance gain is achieved. The best behaving combination corresponding to a deep architecture trained with the end-to-end approach and an NMS threshold of 0.2 for both novel classes (AUCs of 0.84,599 for “cow” and 0.79,612 for “train”). Although many of the achieved accuracy metrics are considerably similar, this set of parameters managed to correctly identify the most instances. Thus, it is appropriate for a novelty detection system operating on greyscale imagery.

4.5 Classification

With the view to measuring the effect of the parameters selected above on the classification accuracy, we evaluate the resulting CN and compare its performance with the original set reported in [11]. Once more, since the existent Mars/Mars-like data sets are not sufficient for training a CNN architecture, we restrict our evaluation to a greyscale equivalent of PASCAL VOC 2007 using every available data set class. In accordance with the original Faster R-CNN paper, the Average Precision2 (AP) metric was utilised in order to compare the respective performances. As can be seen in Table 3, the results are similar to the original implementation, proving that our parameterisation negligibly affects the CN performance and allows its utilisation over the second step of the proposed novelty detection pipeline.

5 CONCLUSIONS

The authors have presented the first CNN-based saliency detection algorithm in the context of robotic space

---

2Average Precision is computed by the AUC of a Precision-Recall graph.
TABLE 3  Average precision metrics (%) comparing the classification performance between the parameterisation proposed in Section 4.4 and the one reported in [11] (reprinted with permission). Both experiments were conducted on the VOC2007 data set using VGG16 architecture.

<table>
<thead>
<tr>
<th>Parameter Setup in Section 4.4</th>
<th>Average</th>
<th>Plane</th>
<th>Bike</th>
<th>Bird</th>
<th>Boat</th>
<th>Bottle</th>
<th>Bus</th>
<th>Car</th>
<th>Cat</th>
<th>Chair</th>
<th>Cow</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>70.7</td>
<td>73.1</td>
<td>78.9</td>
<td>67.5</td>
<td>57.9</td>
<td>52.7</td>
<td>80.8</td>
<td>83.1</td>
<td>83.3</td>
<td>51.3</td>
<td>77.1</td>
</tr>
<tr>
<td>Table</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dog</td>
<td>69.2</td>
<td>79.6</td>
<td>82.6</td>
<td>74.0</td>
<td>73.9</td>
<td>40.0</td>
<td>67.8</td>
<td>67.8</td>
<td>78.3</td>
<td>74.2</td>
<td></td>
</tr>
<tr>
<td>Horse</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bike</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Person</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Plant</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sheep</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sofa</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Train</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TV</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Results reported in [11]</th>
<th>Average</th>
<th>Plane</th>
<th>Bike</th>
<th>Bird</th>
<th>Boat</th>
<th>Bottle</th>
<th>Bus</th>
<th>Car</th>
<th>Cat</th>
<th>Chair</th>
<th>Cow</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>69.9</td>
<td>70.0</td>
<td>80.6</td>
<td>70.1</td>
<td>57.3</td>
<td>49.9</td>
<td>78.2</td>
<td>80.4</td>
<td>82.0</td>
<td>52.2</td>
<td>75.3</td>
</tr>
<tr>
<td>Table</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dog</td>
<td>67.2</td>
<td>80.3</td>
<td>79.8</td>
<td>75.0</td>
<td>76.3</td>
<td>39.1</td>
<td>68.5</td>
<td>67.3</td>
<td>81.1</td>
<td>67.6</td>
<td></td>
</tr>
<tr>
<td>Horse</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bike</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Person</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Plant</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sheep</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sofa</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Train</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TV</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
exploration. While traditional novelty detection approaches require (a) the manual design of a detection mechanism and (b) a disconnected classification step, we utilised the components of a CNN architecture to create a unifying and end-to-end trainable system. Due to lack of a significantly large, real-life Mars data set, we tested the performance of our system employing Earth training and testing data. The evaluation of our system implied that the goal of identifying novel, that is, never seen before, salient areas is feasible. This shows that the proposed system not only is appealing for future space missions, but for any other remote autonomous operation as well. Finally, stretching the limits of our system, we explored the—as is—applicability of our system in Mars-analogue testing data sets. The results are promising, pointing out the need for further experimentation on a large annotated extra-terrestrial planetary data set [21], with training (and pre-training) images significantly different than the testing ones.

As part of our future work, we plan to address our proposal’s deployment aspects on space-qualified hardware. Even though an approach based on deep learning may seem intensive for such missions, recent advancements in the field of artificial neural networks have resulted in model architectures which are specifically designed to operate on mobile devices, respecting their limitations for memory availability and power consumption [28,29]. Based on our experimentation with the shallow model of ZF, our system is not directly influenced by the network’s complexity, allowing the adaptation of such solutions. Furthermore, modern literature contains several instances of network modules that have been successfully transcoded over Field-Programmable Gate Arrays (FPGAs) [30–33]. Such power-conservative units are a typical choice for space exploration missions and provide a valuable alternative for introducing deep learning [34] and Faster R-CNN within NOAH activity.

ACKNOWLEDGEMENTS

This work was supported by the “Novelty or Anomaly Hunter” (NOAH) project, European Space Agency (ESA), Reference No. AO/1-8627/16/NL/LvH. The learning process of the proposed networks was carried out on hardware donated by NVIDIA’s GPU Grant programme.

REFERENCES


How to cite this article: Bampis I., Gasteratos A., Boukas E. CNN-based novelty detection for terrestrial and extra-terrestrial autonomous exploration. IET Cyber-Syst. Robot. 2021;1–12. https://doi.org/10.1049/csy2.12013