



Novelty Detection in Images Using Vector Quantization with Topological Learning

Yann Bernard, Nicolas Hueber, Bernard Girau

► To cite this version:

Yann Bernard, Nicolas Hueber, Bernard Girau. Novelty Detection in Images Using Vector Quantization with Topological Learning. ICECS 2020, 27th IEEE International Conference on Electronics Circuits and Systems, Nov 2020, Glasgow/Virtual, United Kingdom. hal-02984427

HAL Id: hal-02984427

<https://hal.univ-lorraine.fr/hal-02984427>

Submitted on 30 Oct 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Novelty Detection in Images Using Vector Quantization with Topological Learning

Yann Bernard

*French-German Research Institute of Saint Louis
Saint-Louis, France
Université de Lorraine, LORIA, UMR 7503
Vandoeuvre-lès-Nancy, France
yann.bernard@loria.fr*

Nicolas Hueber

*European Laboratory on Sensory Intelligence
French-German Research Institute of Saint Louis
Saint-Louis, France
nicolas.hueber@isl.eu*

Bernard Girau

*Université de Lorraine, LORIA, UMR 7503
CNRS, LORIA, UMR 7503
Vandoeuvre-lès-Nancy, France
bernard.girau@loria.fr*

Abstract—Novelty detection is a key component of biological vision systems, where its role is to extract critical elements for the agents survival from the massive amount of information present in his visual environment. Current vision based embedded systems, such as surveillance cameras, are facing similar challenges as they have to handle a significant amount of sensory data, with limited computing power and memory bandwidth available. In order to perform artificial novelty detection in these systems, it is necessary to have a model able to learn the local visual environment without having any prior knowledge. This study explores bio-inspired unsupervised neural networks models, more precisely self-organizing maps, which are good candidates for this task. We present an original approach consisting of performing novelty detection based on vector quantization and topological learning.

Index Terms—self-organizing maps, novelty detection

I. INTRODUCTION

A. Novelty Detection

Apart from being an important feature of the human visual cortex, novelty detection has been widely studied in machine learning for its usefulness in many applications like cybersecurity, healthcare, damage detection and video surveillance [1]. The principle relies on defining a set of data that will be considered as "normal" and where the novelty detection model is trained on. The model is then given the task to discriminate for any new datapoint if it is part of the "normal" class or if it is a new element (an outlier). It should be noted that there is no objective way to discern between an outlier and a "normal" datapoint, therefore what is considered "new" is always subject to interpretation, i.e. how close it has to be to the normal data to be included in it.

In this paper, we are interested in visual novelty detection for embedded systems, with video surveillance as target application. The challenges with this application are the high dimensionality of the data (images), the naturally high variability in the background and the requirement to precisely locate the novelty in the images. This task has the advantage of having a complete and well maintained dataset available called CDNET2014 [2].

The authors thank the French AID agency (Agence de l'Innovation pour la Défense) for funding the DGA-2018 60 0017 contract.

B. Topological vector quantization

Vector quantization (VQ) is a technique that aims at representing a dataset with a codebook. Each codeword in the codebook is representing a cluster of similar datapoints in the dataset. The codebook can then be used to perform clustering or data compression. The positions of the codewords in the input space are generally found during a competition-based training (as in k-means for instance).

Topological learning algorithms are models that are able to create a topology from their codebook. The topology is a network that connects codewords based on their proximity to each other, so that neighbouring codewords in the topology are also close together in the input space.

Models like self-organizing maps (SOM) [3] and growing neural gases (GNG) [4] and their variants are able to both perform VQ and create a topology that defines neighbourhood relationships between their codewords. The self-organizing map starts with a fixed topology (usually a 2D grid) and "projects" the dataset on this topology during learning. For growing neural gases, the topology is created in the learning phase, so that the neural gases topology closely mimics the datasets shape in the input space. In this paper we will present our novelty detection methods for self-organizing maps, but which could also be applied to all models that perform both vector quantization and topological learning.

Many variants of the SOM exist in the literature. One version of particular interest here is the Cellular SOM [5]. In addition to its faster convergence time, it is designed to take into account hardware constraints. Using a cellular computing approach, CSOM can be efficiently implemented into embedded systems as CSOM neurons only need to interact with immediate neighbouring nodes, thus inducing the potential of massively parallel and scalable computing. The methods of novelty detection presented in this paper are compatible with CSOM and they can be combined into an efficient SOM based novelty detection in embedded systems.

In the following sections, we will present how SOMs work, and how to learn from images. Then we will present two methods for novelty detection in images, one using VQ by SOM, and the other the SOM's topology. We will finish with experimental results, and a discussion about further work.

II. SELF-ORGANIZING MAPS

A. Learning algorithm

A self-organizing map is a discrete map that spatially arranges neurons (or nodes), usually in a two-dimensional grid. Each neuron n is also connected to the input and has a weight vector w_n , or codeword, whose dimension is equal to the input dimension. First, all codewords are initialized with random weights. The training of the SOM lasts a certain number of epochs (100 in our case). One epoch includes enough training iterations such that the whole training dataset has been used once for learning. For each training iteration, a training vector v is picked among the inputs. The best matching unit (BMU) g is then found, it corresponds to the neuron with the minimal L^2 distance between w_g and v . Then the weights of all neurons are updated according to the following equation:

$$w_i(t+1) = w_i(t) + \epsilon(t) \cdot \Theta(\sigma(t), d_{i,g}) \cdot (v - w_i(t)) \quad (1)$$

with $d_{i,g}$ is the normalized distance between node i and the BMU in the topology (not in the input space). Θ is a normalized centered Gaussian function with standard deviation $\sigma(t)$. $\sigma(t)$ is a parameter that influences the neighbourhood function. The higher it is, the more the BMU influences other neurons. In our experiments, we have set it to start at 0.5 and linearly decrease to a final value of 0.001, so that at the beginning of the training all neurons are significantly influenced by the BMU (unfolding the SOM), and at the end, nearly none except the BMU are optimizing the quantization. $\epsilon(t)$ is the learning parameter, it starts at 0.6 and linearly decreases to a final value of 0.05.

B. Learning images

The SOM has first to learn its visual environment, or the scene background in order to perform the novelty detection. The background can be represented by an image, or a sequence of images considered as the "usual" visual environment. These images first have to go through a tiling, or mosaic, process before learning or reconstruction by the SOM. Each image is tiled into smaller sub-images with a parametrable size usually ranging from 5 by 5 pixels to 30 by 30 (all sizes being square for convenience). These sub-images are meant to represent local features of the image. They are then flattened into vectors and used as the input data for the SOM. For instance a 420 by 360 image with 20 by 20 sub-images will be tiled into 378 sub-images (18 rows and 21 columns), and generate the same amount of input vectors for the SOM with sizes of 400 (20×20) if it is a grayscale image, or 1200 if it is RGB.

Once we have those input vectors, we can train the SOM in the usual way. This will result in the SOM's codewords being a representation of the features of the learned image. Once trained, the learned image can be reconstructed with only the codebook, and the list of the best matching unit indices associated to each sub-image location. By replacing each sub-image by the codeword of its the best matching unit, we obtain a reconstructed version of the image. This process is commonly used in lossy image compression with self-organizing maps [6].

III. NOVELTY DETECTION WITH VECTOR QUANTIZATION

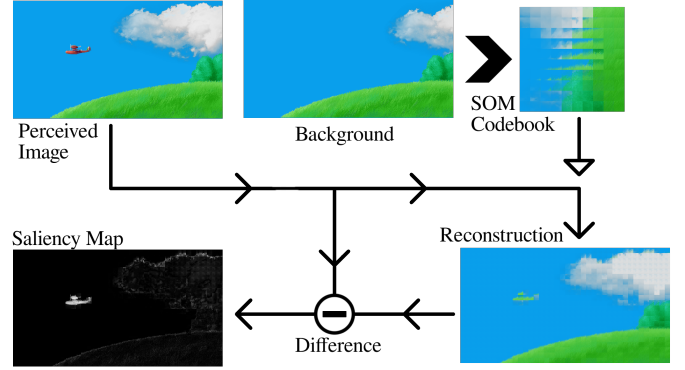


Fig. 1. The process of novelty detection with vector quantization. The SOM is shown with all their codewords put together at their topological places. We can observe that there has been two changes between the background and the perceived image : a plane has appeared and the clouds have moved. The clouds, already present in the background are well reconstructed, the plane however is new, and is not well reconstructed, therefore the difference between the perceived image and the reconstruction highlights the plane more than the clouds (contrary to a simple difference between background and perceived image, where both would be prominent).

In this section, we present the first unsupervised novelty detection method explained with more details in [7]. In order to perform novelty detection with vector quantization, we use the difference between the sub-images of the new image from the sensor and the corresponding best matching unit codewords found in the codebook of the SOM learned on the background image. There are two cases to consider; either the new image from the sensor is similar to any portion of the trained background, or something new is present in the image. In the first case, by reconstructing the new image with the previously trained SOM and comparing it with the new image, the learned codewords should be close to the new image's sub-images (especially if the camera is static). Therefore the difference between the new image and the reconstructed image should be minimal (but still present because of lossy compression and natural changes in the visual environment). However in the second case, we expect the image to be well reconstructed except where the novelty appeared. Because the novelty is by definition something that was not in the background and therefore something that the SOM has not learned. Consequently the sub-images where the novelty is present should be far from their best matching unit's codeword. This in turn will lead to a significant difference when performing the difference between the new image and the reconstructed new image at the place of the novelty. This process is shown in figure 1. This process has the advantage of being precise at the pixel level in highlighting changes, and location independent for novelty detection. However it can be noisy due to imperfect VQ learning and natural variability in the environment.

IV. NOVELTY DETECTION WITH TOPOLOGY

The new approach we propose is based on the topological properties of the learned map to find novelty in an image.

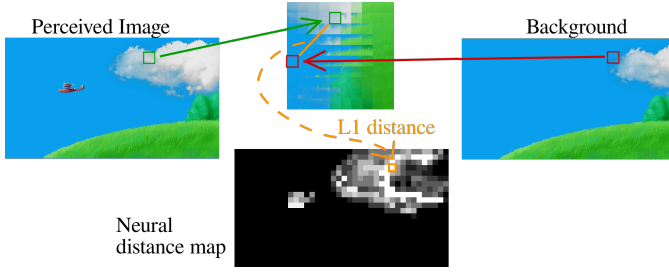


Fig. 2. Topological novelty detection. The process shown here is for one position in the image, and it is repeated on the whole image.

We presented in section I-B how topology links the different codewords of the SOM codebook together by closeness: close neurons in the map topology learn similar codewords. We can use this property to detect novelty in a complementary manner to the method we have described above.

After learning the SOM, we memorize the list of the positions in the map of the best matching units found for all sub-images. When a new image is presented from the sensor, we perform the reconstruction process as in the first novelty detection method, but we are also interested in the list of positions in the map of the best matching units found for all sub-images of the new image. By comparing the two lists we can find changes in the selected BMU positions (in the map) for each sub-image location (in the image). If the BMU is the same between the background and the new image at one location, then there is probably no novelty at this place. If there is a difference in the two BMUs, then there may be some novelty at this position. To quantify the difference between the two BMUs we can compute the topological distance between them in the map. The topological distance is defined by the number of nodes that are crossed in the shortest path between the two BMUs in the map topology. In the classical grid-based SOM, it is simply the L1 distance.

With these distances we can create a saliency map where high distances in the topology mean significant changes in the image, because closeness in the topology means closeness in the input space. This method results in a more robust saliency map, with less level of "noise". But contrary to VQ novelty, the precision is limited to the sub-images size and there is no inhibition of already known elements that have moved. The process is shown in figure 2.

V. EXPERIMENTAL EVALUATION

A. Combination of VQ and Topology Novelty detection

The two above methods have different and complementary properties in their novelty detection. We combined them to obtain the result we show in this section. The combination works by normalizing the two issued saliency maps and multiplying them together to obtain the final saliency map. This straightforward operation does not require any hyper-parameter. Furthermore the new fused model architecture is computationally efficient as it shares the same SOM codebook and the same reconstruction.

B. Experimental protocol

Experiments were performed on the CDNET14 database. It is featuring a varied number of situations and environments that are particularly challenging for novelty detection. We removed three categories from the dataset (*Pan Tilt Zoom*, *low FPS* and *Intermittent object motion*) because one or more of their videos featured a behavior that is not compatible with our novelty detection task (i.e. where the object to be detected is already present at the start of the sequence and part of the background). We generated a background image for each video by computing the median pixels of the sequence so that the model is trained without perturbations from the targets.

For each video, we used a single 20 by 20 SOM with 20 by 20 sub-images. The saliency map generated by our model is binarized with a threshold at 10 (all pixels over 10 are positive). These values were set after optimizing for the global best F-measure. It should be noted that there is a lot of variability between the videos, and optimizing those parameters for each video can lead to a significant performance improvement (for instance some pedestrian videos have better results with smaller sub-images, while other car heavy videos work best with bigger sub-images).

C. Results on CDNET's change detection dataset

Categories	Precision	Recall	F-measure
Baseline	0.6210	0.6382	0.6148
Bad weather	0.3710	0.4712	0.4002
Camera jitters	0.4241	0.5550	0.4706
Dynamic background	0.1008	0.5776	0.1633
Night videos	0.3342	0.4046	0.3423
Shadow	0.5135	0.5206	0.4968
Thermal	0.5607	0.6031	0.5546
Turbulence	0.1063	0.6249	0.1374
Global	0.3790	0.5494	0.3975

TABLE I
MEASURED PERFORMANCE ACROSS DIFFERENT CATEGORIES.

Measured results on the different categories of CDNET are shown in table I and figure 3 shows the results of our novelty detection on different examples. F-measure is a metric regrouping precision (number of true positives over all positives) and recall (number of true positives over true positives and false negatives) with the following formula:

$$\text{F-measure} = \frac{(2 \times \text{Precision} \times \text{Recall})}{\text{Precision} + \text{Recall}} \quad (2)$$

The values displayed in table I are an average result of 10 runs. Therefore the F-measure shown may seem inconsistent with the precision and recall values and equation 2, as the average F-measure is different from the F-measure of the averages of precision and recall.

We can observe that precision is always the limiting factor of performance, especially in the *Dynamic background* and *Turbulence* categories. Both of these having significant and continuous variations from the initial background, which in turn causes a lot of false positives (as shown in figure 3). Recall is ranging from 40% to 65%, which seems to indicate

that approximately half of the pixels that should be positive are found. However, the objects that we are able to detect often are not completely overcast, and with a lot of black spot inside detected objects. This implies that we actually detect more objects than what the recall indicates, as some part of false negatives result from already detected objects that are not well detoured.

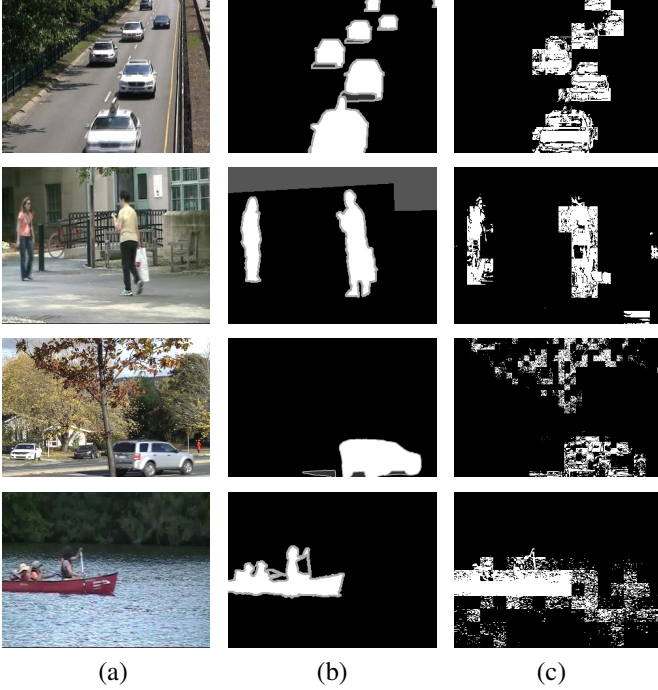


Fig. 3. Results on different scenarios from CDNET. (a) is the input, (b) the groundtruth and (c) our model. We can see the sub-images we use in our novelty detection appear in the resulting saliency map, which gives it its "blocky" look. The novel objects are mostly detected although some of their parts are missing. We can also observe that the model struggles when the background is too changing.

VI. CONSIDERATIONS FOR EMBEDDED SYSTEMS USE

Self-organizing maps are known for their high computational cost, most of it coming from the search for the BMU where all neurons in the map have to be evaluated. However SOMs are naturally parallelisable, and it is thus possible to drastically reduce the time necessary for an iteration. Recent developments presented in [5] show that the use of a cellular computing approach can lead to significant benefits for embedded SOMs, like faster convergence time which in turn makes online learning much more feasible. In CSOM, when the BMU is identified, weight updates propagate in a cellular way from the BMU so that each neuron is "influenced" by the weight vector of a neighbouring neuron that has been updated before, instead of getting closer to the input vector. Current versions of CSOM are cellular and parallel: several input vectors are simultaneously learned, each cellular weight update process being initiated at the level of the corresponding BMU. The hardware device targeted by CSOM is a multi-FPGA Self-configurable Cellular Adaptive Platform (SCALP) [8].

In [9], the authors are presenting an algorithm for searching the BMU with the help of the SOMs topological continuity with a lower complexity than the traditional exhaustive search. The principle is to follow a path from any neuron to the BMU by "descending" the computed distances between the input and each neuron weight vector. A similar approach can be used in a cellular way for our application, as inspired by CSOM: several sub-images (input vectors) can be handled simultaneously, sent in a unicast way to the corresponding stored BMU for the initial background image, and each BMU initiates a descending path towards the BMU of the corresponding sub-image in the new image. Following this approach, several topological distances (length of the descending paths) can be computed simultaneously, and with a minimized risk of communication bottlenecks.

VII. CONCLUSION

We have presented an original approach to novelty detection using both vector quantization and topological learning properties of SOMs. Contrary to typical change detection algorithms, these methods rely exclusively on unsupervised learning and are able to perform without any prior knowledge. Our methods could also be applied to any model performing both vector quantization and topological learning like growing neural gases and its variants. However further improvements and the addition of post-processing are needed in order to match state of the art performance on this application (which can be found on CDNET's website). Further work in improving the hardware implementation of SOMs and cellular computing would benefit our approach, potentially leading to a completely cellular embedded system performing novelty detection and that could be used as a base to more complex bio-inspired visual processing.

REFERENCES

- [1] Marco A.F. Pimentel, David A. Clifton, Lei Clifton, Lionel Tarassenko. **A review of novelty detection**. Signal Processing, 2014
- [2] Yi Wang, Pierre-Marc Jodoin, Fatih Porikli, Janusz Konrad, Yannick Benezeth and Prakash Ishwar. **CDnet2014: An Expanded Change Detection Benchmark Dataset**. IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2014
- [3] Marie Cottrell, Madalina Olteanu, Fabrice Rossi, Nathalie N. Villalaneix, **Self-Organizing Maps, theory and applications**, Revista de Investigacion Operacional, 2018
- [4] Bernd Fritzke. **A Growing Neural Gas Network Learns Topologies**, Advances in Neural Information Processing Systems 7, 1995
- [5] Bernard Girau, Andres Upegui. **Cellular Self-Organising Maps - CSOM**. 13th International Workshop on Self-Organizing Maps and Learning Vector Quantization, Clustering and Data Visualization, 2019
- [6] Christophe Amerijckx, Jean-Didier Legat, Michel Verleysen. **Image Compression using self-Organizing Maps**. Systems Analysis Modelling Simulation, 2003
- [7] Yann Bernard, Nicolas Hueber, Bernard Girau. **Novelty detection with self-organizing maps for autonomous extraction of salient tracking features**. 13th International Workshop on Self-Organizing Maps and Learning Vector Quantization, Clustering and Data Visualization, 2019
- [8] Fabien Vannel, Diego Barrientos, Joachim Schmidt, Christian Abegg, Damien Buhlmann and Andres Upegui. **SCALP: Self-configurable 3-D Cellular Adaptive Platform**. Proceedings of the 2018 IEEE Symposium Series on Computational Intelligence (SSCI), 2018
- [9] Yann Bernard, Nicolas Hueber, Bernard Girau. **A fast algorithm to find Best Matching Units in Self-Organizing Maps**. 29th International Conference on Artificial Neural Networks, 2020, In Press