Unsupervised Feature Selection for Multi-class Object Detection Using Convolutional Neural Networks

Masakazu Matsugu[1] and Pierre Cardon[2]

[1]Canon Inc. HVS Research Dept. 5-1, Morinosato-Wakamiya, Atsugi, 243-0193 Japan
matsugu.masakazu@canon.co.jp
[2] EU-Japan Center for Industrial Cooperation, 13-3, Ichibancho, Tokyo, 102-0082 Japan

Abstract. Convolutional Neural Networks (CNN) have proven to be useful tools for object detection and object recognition. They act like feature extractor and classifier at the same time. In this study we present an unsupervised feature selection procedure for constructing a training set for the CNN and analyze in detail the learnt receptive fields. We then introduce, for the first time, a figural alphabet to be used for low-level feature detection with CNN. This alphabet turned out to be useful in detecting a vocabulary set of intermediate level features and considerably reduces the complexity of the CNN. Moreover we propose an optimal high-level feature selection procedure and apply this to the challenging problem of car detection. We demonstrate promising results for multi-class object detection using obtained figural alphabet to detect considerably different categories of objects (e.g., faces and cars).

1 Introduction

In this work, we address the problem of selecting optimal local features for multi-class object detection [10]. A crucial aspect for object detection is the choice of an optimal set of features. In [2] and [11], an interest point operator and a k-means clustering algorithm are used to extract and regroup high-level features for estimating the parameters of the underlying probabilistic model. In [5], the image entropy is adopted to select interesting areas in the image and a Self-Organizing Map (SOM) [6] to organize the big amount of extracted high-level features, then a clustering algorithm is used to regroup similar units in the SOM to an automatic determined number of macro-classes. In [9], sub-optimal features are selected, for training the convolutional neural networks (CNN), by trial and error and extracted manually.

CNN [8] as well as *neocognitrons* [3] have been used for face detection and recognition [9], [7]. In [9], a variant of back-propagation algorithm is proposed to teach each layer separately (sequential BP: SBP) so that the extracted features are controlled, and also some specific parts of the face can be detected. The first two layers are trained with intermediate-level features (e.g. eye-corners), while the subsequent layers are trained with more complex, high-level features (e.g. eyes, faces...). This requires to select a training set of features. By selecting a limited set of

F. Yin, J. Wang, and C. Guo (Eds.): ISNN 2004, LNCS 3173, pp. 864-869, 2004.

features for a specific object, we may expect to find a restricted yet useful set of receptive fields as in neurophysiological studies [1], [4].

In this work, we present an unsupervised feature extracting and clustering procedure, using an interest operator combined with a SOM (Section 2). This method combines the advantages of both [11] and [5] by selecting a limited number of features and regrouping them using a topographic vector quantizer (SOM); acting like a vector quantizer and introducing a topographic relation at the same time.

The obtained feature classes are self-organized, low-and intermediate-level features that are used to train the 2 first layers of the CNN and obtain a minimum set of 4 alphabetical receptive fields by back-propagation (Section 3). This alphabet considerably reduces the complexity of the network by decreasing the number of parameters and can be used for detection of different object classes (e.g. faces, cars,...). We also introduce a method to select optimal high-level features and illustrate it with the car detection problem (subsection 3.3).

2 Unsupervised Local Feature Extraction in CNN

We use a modified architecture of CNN [9], which inherits classical architecture with shared weights, local receptive field and subsampling layers. The whole network is described in the lower part of Fig.1.

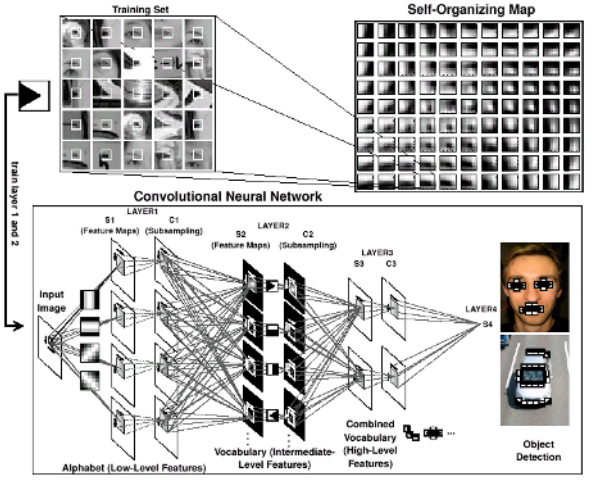

Fig. 1. CNN model (lower half) for multi-class object detection and alphabetical local features (upper right) obtained from SOM with training set (upper left)

Some specific local fragments of image extracted a priori, by using the proposed method in this study, are used to train the first two layers of the CNN. First, we train the CNN to recognize only one feature (one output plane in S2). A sequential back-propagation algorithm [9] is used for learning and weights are updated after each training pattern (fragments of images) is presented. A fixed number of 100 epochs has been used. For each training set, a different number of cell-planes in layer S1 have

been tested. The network has essentially four distinct sets of layers: S1-C1, S2-C2, S3-C3, S4 (S_k: the kth feature detecting layer; C_j: the jth feature pooling layer for subsampling). Layers S3-C3 and above are concerned with object specific feature detection. In order to limit the number of features to object-relevant features, an interest point operator is used. This operator selects corner-like features in the image.

Having selected a restricted number of points we extract features around these points. These features are used as learning set for the SOM well suited for classifying and visualizing our feature set. It turned out that the illumination has a big influence on the classification of our features, so we have rescaled the feature set between -1 and 1 before applying the SOM. Each unit of the SOM defines a training set for the CNN. Once lower-level alphabetical feature detectors are formed, higher level feature detectors can be obtained from BP with connections between neurons below intermediate layers fixed.

3 Results

3.1 Unsupervised Selection of Low Level Features

Since we are interested in low-level features to train the first two layers of the CNN, we have chosen to extract small (7 x 7) features. With a database of 904 (size: 208 x 256) images (300 faces (frontal view), 304 cars (upper view) and 300 various types of images), we obtained a set of 69,753 features.

We start by manually selecting units that have a simple character (horizontal, vertical and diagonal contrast). The SOM has been calculated using the SOM Toolbox in Matlab. We have fixed the umber of units to 100, based on the assumption that there are not more than 100 different types of local (7 x 7) features in an object image appropriately cropped so that irrelevant background features are cut out.

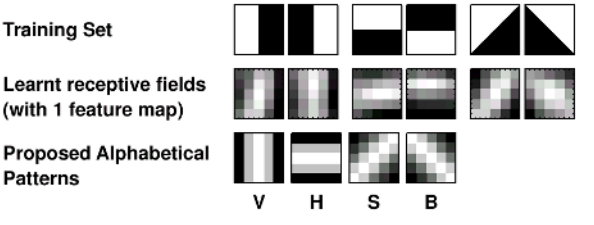

Fig. 2. Learnt receptive fields for low-level features and corresponding alphabetical patterns

For each cluster we only consider the 300 features, which are the closest to the SOM-unit, in terms of Euclidean distance. 200 features are used for training, 50 features for validation and the last 50 units for testing. The results have been obtained with a test set of 50 features and optimal receptive fields have been selected by cross-validation. We see that for such simple features, only one cell-plane in S1 is sufficient to obtain good detection results. We also notice that the learnt receptive fields (Fig.2) have a regular pattern. Based upon these patterns we propose a set of 4 alphabetical patterns ***V***, ***H***, ***S***, ***B*** (hereafter, represents vertical, horizontal, slash, backslash, respectively) described in Fig.2.

3.2 Intermediate Level Features as Vocabulary

We observe that some feature clusters in the SOM have a more complex aspect as shown in Fig.3. We claim that these more complex features can be detected using the simple receptive fields, described in the previous section.

Considering for example the feature described in Fig.3, we see that this eye-corner type feature can be decomposed into 2 local alphabetical features. After training the CNN to detect this type of feature, we obtain the results with FAR: 0%, 4%, 8%, and 10% for ***H***, ***V***, ***S***, and ***B***, respectively.

The usefulness of our alphabetical set appears when we want to detect several high-level features with a small number of receptive fields, using synergies between the features. Let us consider the features used to detect a complete eye or a mouth [9]. They can be decomposed to 2 horizontal, 2 slash and 2 back-slash components (Fig.4).

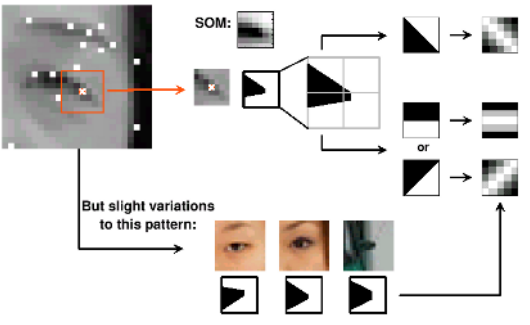

Fig. 3. Intermediate-level feature (eye-corner) decomposed in low-level features

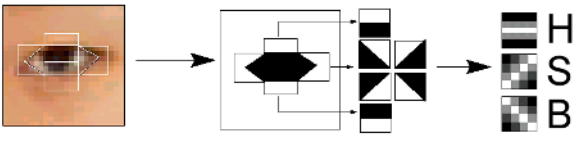

Fig. 4. High-level feature (i.e., eye) for face-detection, decomposed into low-level features

With a limited set of three fixed receptive fields ***H***, ***S*** and ***B*** it turned out that we reach a detection rate of eye-corner comparable to that of using six learnt receptive fields. Our alphabetical set, being close to the optimal set of weights, therefore outperforms the learnt weights. We can extend these results for different types of complex features and construct a vocabulary set that can be recognized with ***H***, ***V, S***, and ***B***. For illustration purposes, we have tested our alphabet with images from which features have been extracted. It turned out that we could detect, in the S2 layer, eye- and mouth-corners as well as the side mirrors of a car, using only three receptive fields (***H***, ***S*** and ***B***).

3.3 Extraction of High Level Features Using Alphabetical Features

A last question to be answered is which vocabulary we should use, in other words, what features are important to detect a specific object. To find these features we apply classical BP (hereafter referred as GBP: global BP), not the proposed SBP, to the entire CNN with connections below S3 layer (S1-C1-S2-C2) fixed, and analyze the output of Layer3 (high-level features). The GBP converges to a local minimum, therefore the algorithm will tend to extract sub-optimal features to minimize the detection error.

To examine the validity of our scheme, we applied our method to a training set of images of bright-colored cars with significant variance in shape, illumination, size and orientation. The size of the images used for learning was 156 x 112, and 90 images were used for training and 10 images for validation. We aim to find characteristic high-level features for the detection of this type of cars and for this particular view. In addition, we need to tailor our model to be able to distinguish between cars and other rectangular objects. So, we have included a set of negative non-car examples, with similar rectangular shape but which were not cars.

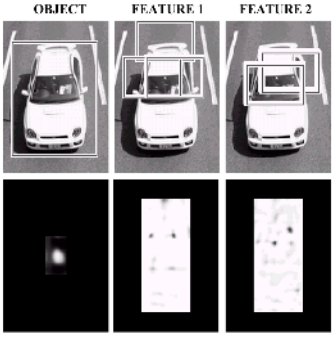

Fig. 5. (a) Car detection in the top layer, (b),(c)intermediate detection results for feature1 and 2

We used our figural alphabet between the input layer and S1 (fixed), and obtained other receptive fields by GBP. Using the four receptive fields ***H***, ***V***, ***S***, and ***B***, the number of cell-planes in S1-C1, S2-C2, S3-C3 and S4 were respectively 4, 3, 2 and 1. Respective sizes of the receptive fields were 5 x 5(Input-S1), 9 x 9(C1-S2), 25 x 45(C2-S3), and 79 x 29(C3-S4). After a fixed number of 500 epochs, we selected the receptive fields by cross-validation. With these receptive fields, we obtained a detection rate of 72% (test set of 50 cars) and 13 false alarms for 100 non-car images.

Having discovered the important features for our object detection problem, we obtain object specific vocabulary to select to construct these high-level features. We can use SBP as in [9] to train the higher level layers in the CNN: to train layer by layer with the selected vocabulary features. By doing so we limit the computing time which can be very long in the case of GBP and obtain very feature specific detectors in each layer. Typical detection result is shown in Fig.5.

4 Summary

We have proposed an automatic feature extraction procedure combining, for the first time, an interest point operator [11] with a SOM [5] to extract a training set of features for the CNN. By training the CNN with this training set, we found figural alphabets of 4 simple receptive fields obtained by SBP [9] are good enough to detect frontal view of cars as well as faces. We have shown that in spite of the simplicity of this alphabet it gives remarkable results, comparable and sometimes better than the learnt receptive fields with average detection rate over 95% for different types of features (see subsection 3.2). After obtaining alphabetical feature detectors in the S1 and S2 layer of CNN, we applied GBP to the S3 and S4 layers of CNN, with lower level weights fixed, to obtain higher level feature detectors (e.g., cars and faces), thereby obtaining sub-optimal vocabulary set. The optimality was examined in terms of cross-validation. In summary, we showed that the proposed method can be used to extract useful, generic local features for multi-class object detection (e.g., face and car detection) in the framework of convolutional neural networks.

References

1. Blackmore, C., Cooper, G. E.: Development of the Brain Depends on the Visual Environment. Nature, 228 (1970) 477-478
2. Burl, M., Leung, T., Perona, P.: Face Localization via Shape Statistics. In: Intl. Workshop on Automatic Face and Gesture Recognition (1995)
3. Fukushima, K.: Neocognitron: A Self-organizing Neural Network Model for a Mechanism of Pattern Recognition Unaffected in Shift Position. Biol. Cybern., 36 (1980) 193-202
4. Hubel, D., Wiesel, T.: Receptive Fields, Binocular Interaction and Functional Architecture in the Cat's Visual Cortex. Journal of Physiology, 160 (1962) 106-154
5. Ikeda, H., Kashimura, H., Kato, N., Shimizu, M.: A Novel Autonomous Feature Clustering Model for Image Recognition. In: Proc. of the 8th International Conference on Neural Information Processing (2001)
6. Kohonen, T.: Self-organizing Maps. Springer-Verlag, Berlin (1985)
7. Lawrence, S., Giles, G. L., Tsoi, A. C., Back, A. D.: Face Recognition: A Convolutional Neural Network Approach. IEEE Transactions on Neural Networks, 8 (1995) 98-113
8. Lecun Y., Bengio, Y.: Convolutional Networks for Images, Speech, and Time-Series. In: Arbib, M. (ed.): Handbook of Neural Networks and Brain Sciences, MIT Press, Cambridge (1995) 255-258
9. Matsugu, M., Mori, K., Ishii, M., Mitarai, Y.: Convolutional Spiking Neural Network Model for Robust Face Detection. In: Proc. of the 9th International Conference on Neural Information Processing (2002) 660-664
10. Papageorgiou, C. P., Oren, M., Poggio, T.: A General Framework of Object Detection. In: Proc. of International Conference on Computer Vision (1998) 555-562
11. Weber, M., Welling, M., Perona, P.: Unsupervised Learning of Models for Recognition. In: Proc. of the 6th European Conference on Computer Vision (2000)