Convolutional neural networks for robot vision: numerical studies and implementation on a sewer robot.

Saeed Shiry
GMD-Japan Research Laboratory,
Collaboration Center,
2-1 Hibikino, Wakamatsu-ku,
Kitakyushu-city
saeed.shiry@gmd.gr.jp

Matthew Browne
matthew.browne@gmd.gr.jp

Abstract

Convolutional neural networks (CNNs) impose constraints on the weights, and the connectivity of a standard neural network, providing a framework well suited to the processing of spatially or temporally distributed data. Although CNNs have been applied to face and character recognition, they have still received relatively little attention. The present paper applies the CNN architecture to an artificial test problem and to an application in robot vision. Autonomous sewer robots must navigate independently the sewer pipe system using information from sensors. One required component is robust detection of pipe joints and inlets using data from the omnidirectional sensor. A simple CNN is shown to robustly classify 32x32 pixel normalized video frame data on a limited validation set. The study indicates that machine learning methods for robot vision are feasible in terms of classification accuracy and online implementation.

1 Introduction

The goal of sewer robotics is to construct an autonomous robot capable of autonomously conducting sewer inspection tasks in sewerage systems [13]. Autonomous sewer robots are robots designed to perform automated inspection of sewer pipe systems. These robots must navigate independently the sewer pipe system using only available information from sensor systems. The task of accurately detecting and classifying relevant landmarks and features in the environment is an essential part of the navigation routines. Because video is often used for the inspection work, performing detection of cracks and other faults, it is useful if the same data can be utilized for landmark detection.

Paletta, Rome and Platz [13] used a multi-stage system for landmark detection based on video data. This involved an attention controller, pre-processing, feature extraction, probabilistic interpretation, and final classification. Early components of the system operate according to fixed *a priori* rules while latter components are data-driven and adaptive. A related work for sewer pipe navigation include procedures for navigating under uncertainty [12].

Other pattern recognition approaches designed for use in autonomous sewer inspection include application of neural architecture to segmentation of pipe joints [7] and reconstruction of a 3D model of the interior of the pipe based on video data [6].

In this work we follow the general viewpoint of [13], holding that interpretation of sensor images should be learnt from experience, and modeling of features of interest takes place in terms of their appearance to the agent. This is therefore a strong data-based perspective: analytical approaches towards constructing an objective model of the objects in question are rejected in favor of methods that directly learn from experienced sensor data. The present work attempts to take this perspective further, in attempting to implement an entirely trainable system. Thus, pre-processing and feature detection, transformation and classification modules are integrated into a single adaptive neural architecture. Convolutional neural networks (CNNs) form the theoretical basis that makes such an approach possible.

2 Convolutional Neural Networks

The term *convolutional network* (CNN) is used to describe an architecture for applying neural networks to two-dimensional arrays (usually images), based on spatially localized neural input. This architecture has also been described as the technique of shared weights or local receptive fields [16, 5, 14] and is the main feature of Fukushima's *neocognitron* [11, 10]. Le Cun and Bengio [4] note three ar-

chitectural ideas common to CNNs: local receptive fields, shared weights (weight averaging), and often, spatial downsampling. Processing units with identical weight vectors and local receptive fields are arranged in an spatial array, creating an architecture with parallels to models of biological vision systems [4]. A CNN image mapping is characterized by the strong constraint of requiring that each neural connection implements the same local transformation at all spatial translations. This dramatically improves the ratio between the number of degrees of freedom in the system and number of cases, increasing the chances of generalization [15]. This advantage is significant in the field of image processing, since without the use of appropriate constraints, the high dimensionality of the input data generally leads to ill-posed problems. To some extent, CNNs reflect models of biological vision systems [9]. CNNs take raw data, without the need for an initial separate pre-processing or feature extraction stage: in a CNN the feature extraction and classification stages occur naturally within a single framework.

In the CNN architecture, the 'sharing' of weights over processing units reduces the number of free variables, increasing the generalization performance of the network. Weights are replicated over the spatial array, leading to intrinsic insensitivity to translations of the input - an attractive feature for image classification applications. CNNs have been shown to be ideally suited for implementation in hardware, enabling very fast real-time implementation [17]. Although CNNs have not been widely applied in image processing, they have been applied to handwritten character recognition [5, 17, 3, 1] and face recognition [15, 9, 8]. CNNs may be conceptualized as a system of connected feature detectors with non-linear activations. The first layer of a CNN generally implements non-linear template-matching at a relatively fine spatial resolution, extracting basic features of the data. Subsequent layers learn to recognize particular spatial combinations of previous features, generating 'patterns of patterns' in a hierarchical manner. If downsampling is implemented, then subsequent layers perform pattern recognition at progressively larger spatial scales, with lower resolution. A CNN with several downsampling layers enables processing of large spatial arrays, with relatively few free weights.

To sum, CNNs perform mappings between spatially / temporally distributed arrays in arbitrary dimensions. They may be applied to time series, images, or video. CNNs are characterized by:

- translation invariance (neural weights are fixed with respect to spatial translation)
- local connectivity (neural connections only exist between spatially local regions)
- an optional progressive decrease in spatial resolution (as the number of features is gradually increased).

Often when applying CNNs we wish to progressively reduce spatial resolution at each layer in the network. For example, a CNN may be used for classification where an image is mapped to a single classification output. Given fixed filter sizes, reducing spatial resolution has the effect of increasing the effective spatial range of subsequent filters. In a CNN with subsampling in each layer, the outcome is a gradual increase in the number of features used to describe the data, combined with a gradual decrease in spatial resolution. Because the change in coordinate system is accomplished in a nonlinear, incremental, hierarchical manner, the transformation can be be made insensitive to input translation, while incorporating information regarding the relative spatial location of features. This provides an interesting contrast to methods such as principle components analysis, which make the transition from normal coordinate space to feature space in a single linear transformation.

Please refer to [2] [1] for a formal description of a CNN. However, a graphical example of the architecture should assist in conceptualizing the operation of a CNN. Fig. 1 show an elementary network that maps 8x8 input arrays to a single output array via three layers, each consisting of a 2x2 weight vector. The feature arrays are formed by convolving the weights vectors with the previous array, using a step size of 2. The network has 64 inputs and 84 connections, but due to the constraint of translation invariance, there are only 12 free weights.

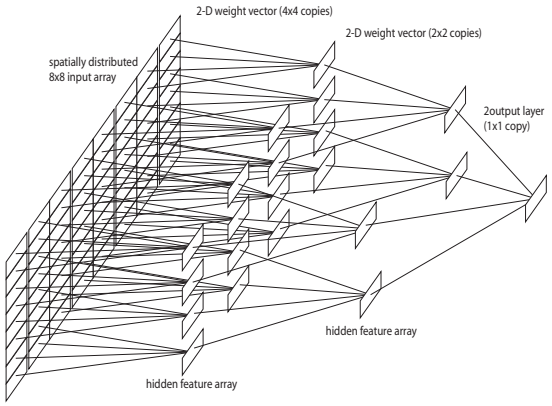

Figure 1. An elementary CNN architecture

3 Numerical tests

Although impressive applied applications of CNNs have been reported (e.g. [1]), less work has been done to specify the theoretical performance of CNN networks, either analytically or through numerical tests. Previous work [2] has

[1] preprint available at http://www.gmd.gr.jp/html/ matthew.browne/

defined and detailed the formal structure of a CNN. Here we shall detail a numerical experiment that is used partly to elucidate the essential properties of CNNs, and partly in order to confirm that CNNs can learn spatially invariant, non-linear filtering systems.

Small 4x4 pixel input arrays were considered. Each input array considered of two 'micro features', either two pixels in top-left to bottom-right (type A) diagonal arrangement, or two pixel in bottom-left to top-right diagonal arrangement (type B). Each micro-feature was allowed to vary independently over all possible spatial locations in the input space. Considering input arrays of two micro-features, four combinations are possible: AA, BB, AB, BA. Figure 2 displays class AA and class BB while figure 3 displays the combined class AB / BA.

The first numerical experiment was to test that a CNN could correctly differentiate between classes AA and BB. Casual inspection of figure 2 would indicate that the various permutations and interference between the micro features creates a non-trivial and perhaps challenging problem for a normal neural network. It was expected that a CNN with 2@2x2:1@3x3 filters and 29 free weights was sufficient to differentiate between the two classes [2]. The network successfully classified all inputs except those shown in the lower section of figure 2. In these cases, interference between the micro features creates identical patterns. Although the CNN can theoretically learn purely spatial differences, sensitivity to pure translation would have violated the underlying decision rule required to classify the rest of the data set, namely orientation differences.

The second numerical experiment was more challenging: the CNN was required to differentiate between class AB/BA and class AA/BB. This represents a kind of spatial X-OR problem, with the added difficulty of interference between features. A 4@2x2:4@3x3 CNN was trained to convergence on this data set. 100% performance was obtained on this problem, demonstrating that a minimal CNN architecture can learn a non-linear translation-invariant mapping in two dimensional feature space.

The key property of the first layer of the CNN is that only a single template is learnt for all possible translations. The following layer integrates information across different templates and spatial translations, enabling approximation of the X-OR function.Thus, *CNN units de-couple detection of a feature's shape and a feature's location*. In conventional spatial feature extraction (principle component 'eigenfaces' are a prominent example) shape and location are clamped. Thus, a separate representation of a feature must be learnt over the entire of range spatial locations where it may occur. Although the artificial problem is highly stylized, realistic detection of complex objects in spatially or temporally distributed data usually involves uncertainty in the absolute and relative position of sub-features.

[2]The first layer consisting of 2 filters of size 2x2 capture the two micro filters, the final filter integrates information over the 3x3 feature space. A total of 29 free weights are used in the network.

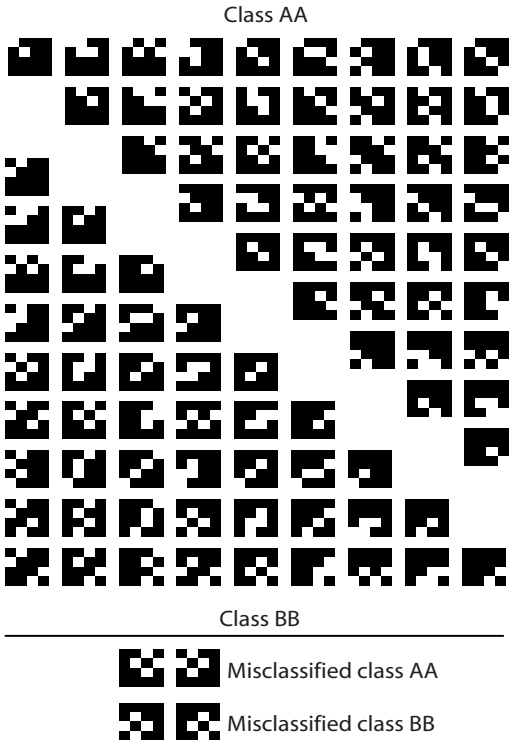

Figure 2. Differentiation of classes 1 and 2 requires translation invariance and tolerance of interference between features.

4 Application in service robotics

The current project involves video data gathered from a robot fitted with an omnidirectional camera with the task of navigating and inspecting civil sewer pipe systems. Effective navigation requires that the robot be capable of detecting pipe inlets and pipe joints using the same sensor data used for fault detection, namely the omnidirectional data. Figure 4 displays examples of each sensor data class. The CCD original camera image frames were truncated and down sampled to arrays of 36x36 pixels. It was found that this was approximately the maximum amount of downsampling that could be achieved while preserving enough resolution for detection of the relatively finely structured pipe

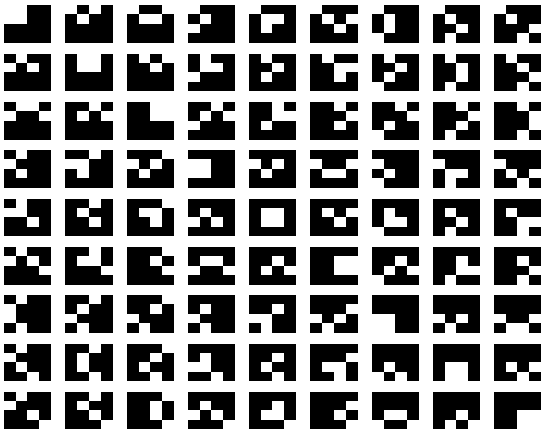

Figure 3. Differentiation of of class 3 from classes 1 and 2 represents a spatial X-OR problem requires implementation of a translation invariant non-linear spatial filter system.

joints. Standard 1:255 valued black and white pixel intensities were normalized across frames to lie within a range of 0:1.

Dirt, reflections, and changes in reflectance represented a challenge in this classification task. However, the objects to be detected had a relatively simple structure. The main challenge for classification of this kind of spatial input by machine learning methods is the size of the input ($36^2 = 1296$ inputs) and variability in the scale and location of features as they pass through the robot's visual field [3].

Table 1 shows the relatively simple architecture used for classification of the camera frames. Only one feature map per layer was used to detect pipe joints and pipe inlets, respectively. The 'inlet detector' and the 'joint detector' subnetworks each consisted of a total 51 free weights, including biases to each 2D weight vector. We note that the input size of the 36x36 was somewhat tailored to the architecture, since application of a 5x5 convolution filter without downsampling results in a 32x32 array, which is convenient for subsampling to a single 1x1 array, corresponding to the class prediction. Tan-sigmoid and log-sigmoid trans-

[3] A conventional neural network would require a correspondingly large number of free weights. Pre-processing methods such as PCA would decrease the number of free weights required, but generally create features highly sensitive to translation and scale

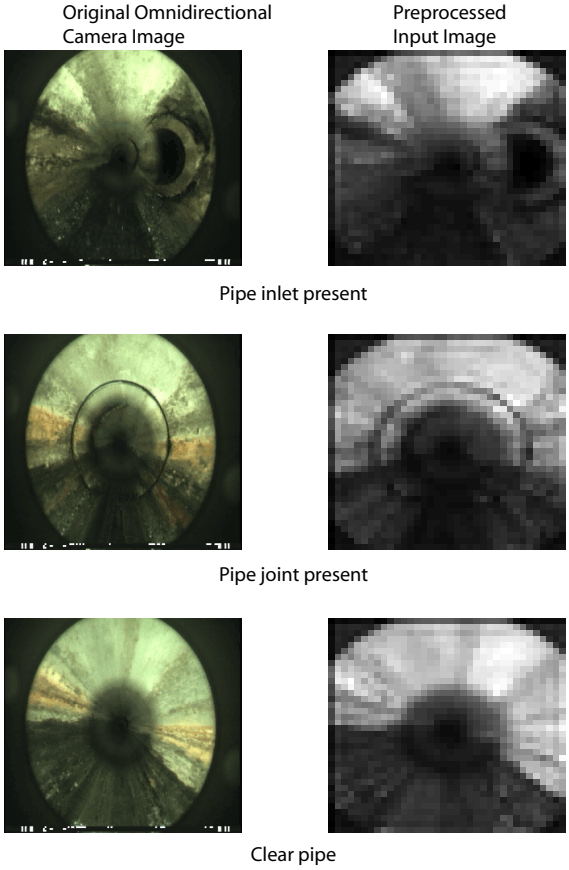

Figure 4. Examples of three classes of omnidirectional camera images for classification by the CNN.

fer functions were used.

Table 1. Architecture of CNN used for omnivision camera processing.

layer	1	2	3	4	5	6
filter size	5	2	2	2	2	2
map size	32	16	8	4	2	1
downsampling	no	yes	yes	yes	yes	no
transfer fun.	tan	tan	tan	tan	tan	log

The networks were trained using 160 manually classified video frames drawn from a sample run through the test pipe, consisting of an approximately equal number of 'joint present', 'inlet present', and 'nothing present' class samples. Training was performed for 1000 epochs, using backpropagation, with momentum and an adaptive learning rate. After training, mean square error rates of 0.0022 and 0.0016 were obtained.

Validation was performed on a second 840 frame, 42 second video sample through the same pipe. For classification of continuous video data, subsequent frames are not statistically independent, and there also exists some 'grey area' where one class stops and the next starts. Thus, calculation of an overall correct classification rate is rather misleading. A more accurate description of the performance of the network is provided in fig. 5, which displays the output of the CNN, along with the actual state of the sewer pipe within the visual field of the robot.

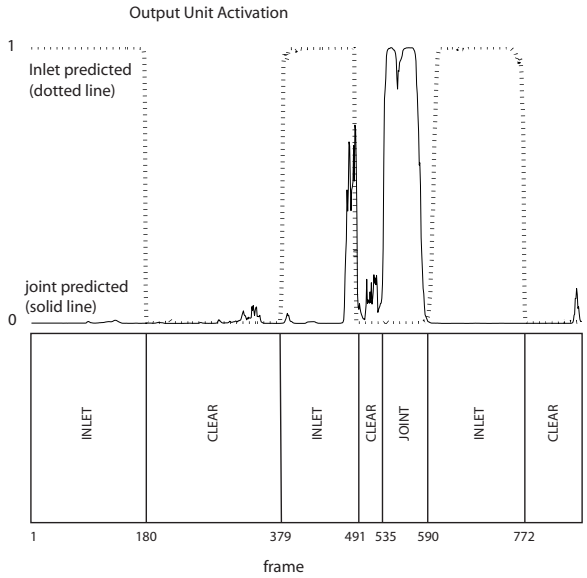

Figure 5. Predicted and actual objects in the visual field of the omnidirectional camera for validation of CNN network.

5 Discussion

The present paper has presented a demonstration of the properties of CNNs using artificial data, and the application of the architecture to an applied problem in robot vision. CNNs are shown to implement non-linear mappings of features with invariance to spatial translation. More precisely, CNNs decouple the learning of feature structure and feature location, and are therefore well suited to problems where the relative or absolute location of features has a degree of uncertainty. It is the contention of the authors that this is a general rule, rather than a special case, in classification of spatially or temporally distributed data.

CNNs have been studied with a view to application in robot vision, where we have found the properties of spatial invariance and weight constraints are necessary for application of machine learning methods to high dimensional image input, where features of interest may occur at a variety of spatial locations. The validation results in figure 5 show that the CNN are very positive indication that CNNs may be effectively applied to detecting navigational landmarks

in the sewer-pipe environment. With appropriate thresholding, the activations of the 'inlet detector' and 'joint detector', would result in perfect landmark detection for the validation data. We note the uncertainty of the 'joint detector' around frame 491 is due to a slightly unusual inlet construction that bears some similarity to a normal pipe joint. The system is computationally efficient, capable of processing image frames using an on-board processor in real time.

With respect to the development of an industrial-standard landmark-detection system, much work is required to train and test the network with a wider variety of environments and lighting conditions. We emphasize that the present results should be treated only as a promising indication of the effectiveness of CNNs for robot vision in sewer robots.

The CNN architecture also requires further development. Although basic gradient descent with an adaptive learning rate is adequate, implementation of more advanced optimization techniques (such as Levenburg-Marquardt or conjugate-gradient optimization) is a priority. The basic CNN framework allows a wide variety of possible network architectures: are currently investigating pruning and growing algorithms for specification of the various network parameters. Finally, although CNNs are an efficient method of applying neural networks to image processing, real-time processing of high definition images with a sophisticated architecture would appear to be computationally infeasible without the use of specialized hardware implementation.

References

[1] Y. Bengio, Y. Le Cun, and D. Henderson. Globally trained handwritten word recognizer using spatial representation, convolutional neural networks, and Hidden Markov Models. In J. D. Cowan, G. Tesauro, and J. Alspector, editors, *Advances in Neural Information Processing Systems*, volume 6, pages 937–944. Morgan Kaufmann Publishers, Inc., 1994.

[2] M. Browne and S. Shiry. Convolutional neural networks for image processing: an application in robot vision. In *to appear at Australian Joint Conference on Artificial Intelligence*, 2003.

[3] Y. L. Cun. Generalization and network design strategies. Technical Report CRG-TR-89-4, Department of Computer Science, University of Toronto, 1989.

[4] Y. L. Cun and Y. Bengio. Convolutional networks for images, speech, and time series. In M. Arbib, editor, *The Handbook of Brain Theory and Neural Networks*, pages 255–258. MIT Press, Cambridge, MA, 1995.

[5] Y. L. Cun, J. Boser, D. Denker, R. Henderson, W. Howard, W. Hubbard, and L. Jackel. Backpropagation applied to handwritten zip code recognition. *Neural Computation*, 4(1):541–551, 1988.

[6] T. P. D. Cooper and N.Taylor. Towards the recovery of extrinsic camera parameters from video records of sewer surveys. *Machine Vision and Applications, 11*, pages 53–63, 1998.

[7] J. R. del Solar and M. K-pen. Sewer pipe image segmentation using a neural based architecture. *Pattern Recognition Letters, 17*, pages 363–368, 1996.

[8] B. Fasel. Facial expression analysis using shape and motion information extracted by convolutional neural networks. In *Proceedings of the International IEEE Workshop on Neural Networks for Signal Processing (NNSP 2002), Martigny, Switzerland*, 2002.

[9] B. Fasel. Robust face analysis using convolutional neural networks. In *Proceedings of the International Conference on Pattern Recognition (ICPR 2002), Quebec, Canada*, 2002.

[10] K. Fukushima. Neocognitron: A hierachical neural network capable of visual pattern recognition. *Neural Networks*, 1(2):119–130, 1988.

[11] K. Fukushima, S. Miyake, and T. Ito. Neocognitron: a neural model for a mechanism of visual pattern recognition. *IEEE Transactions on Systems, Man, and Cybernetics*, 13:826–834, 1983.

[12] J. Hertzberg and F. Kirchner. Landmark-based autonomous navigation in sewerage pipes. In *Proc. First Euromicro Workshop on Advanced Mobile Robots (EUROBOT '96)*, pages 68–73. Kaiserslautern. IEEE Press, 1996.

[13] E. R. L. Paletta and A. Pinz. Visual object detection for autonomous sewer robots. In *Proc. 1999 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '99), vol. 2*, pages 1087–1093. IEEE Press, Piscataway, NJ, 1999. ISBN 0-7803-5184-3.

[14] K. Lang and G. Hinton. Dimensionality reduction and prior knowledge in e-set recognition. In D. Touretzky, editor, *Advances in Neural Information Processing Systems*, pages 178–185. Morgan Kauffman, San Marteo, CA, 1990.

[15] S. Lawrence, C. L. Giles, A. C. Tsoi, and A. D. Back. Face recognition: A convolutional neural network approach. *IEEE Transactions on Neural Networks*, 8(1):98–113, 1997.

[16] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning internal representation by error propagation. In D. E. Rumelhart and J. L. McClelland, editors, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, volume 1, pages 318–362. Cambridge, MA: MIT Press, 1986.

[17] E. Sackinger, B. Boser, J. Bromley, and Y. LeCun. Application of the anna neural network chip to high-speed character recognition. *IEEE Transactions on Neural Networks*, 3:498–505, 1992.