Intelligent Image Recognition System

Victor M. Sineglazov
National Aviation University
Kyiv, Ukraine
svm@nau.edu.ua

Vitaliy Ischenko
National Aviation University
Kyiv, Ukraine
IschenkoVitaly@gmail.com

***Abstract*—The design problem of intelligent image recognition system is considered. For the solution of this problem it is proposed to use constitution neural networks with fuzzy classifier pretraining with the help of the restricted Boltzmann machine.**

Keywords—convolution layer; input layer; output layer; convolution neural network; feature map; UAV

I. Introduction

Since the disturbances presence the traditional navigation systems such as inertial, satellite, aeromagnetometry and others are inoperative. Therefore, there is a necessity to develop a navigation system that is shut out of these drawbacks and is operable in the presence of the radio disturbances.

II. Problem Statement

The given problem solution could be a visual navigation system in which the image recognition process is carried out with the help of artificial neural networks (ANN). Let X is a set of objects descriptions; Y is a set of classes numbers (or names). There is unknown target dependency – reflection y^*: $X \to Y$, values of which is known only on the final objects sample

$$X^m = \{(x_1, y_1), ..., (x_m, y_m)\},$$

where X^m is a set of the training sample elements dimensionality m.

It is required to construct an algorithm able to determine an arbitrary object $x \in X$ belonging to the class $y \in Y$.

III. Visual Navigation Methods Review

Analysis of visual navigation systems, which are based on the methods of computer vision in conditions of electronic warfare (EW) has shown no flexibility, and more often, inability to perform a navigation tasks.

There are two main methods of the navigation problem solution.

1) Refinement of an unmanned aerial vehicle (UAV) location using an images from camera by comparing the given image with the already crosslinked and digitized on-board map. The main drawback of this method or rather a question is where to get a digital map of the EW zone? This is requires the primary UAV that can perform a navigation task with visual navigation system which does not need a digitized on-board map, and will be able to prepare a set of images during the auto mission. Then, with the successful completion of the mission these images must be processed by the ground-based computer and crosslinked into a single map for navigating the rest of the UAVs. In this case the map is actual for a small time, until the real objects form on it hasn't changed and weather conditions are not changed.

2) A self-contained navigation from frame to frame. At the UAV's takeoff the first reference image with the current coordinates are taken. Further images are taken with the time *dt* that provides the current and the previous frames overlapping. It provides an opportunity to find a reference points on the first image and their matches on the second one, forming a milestones' pairs with coordinate's displacement recalculation. The accuracy of this method depends on the on-board computer computing capacity, which is limited. Low region informativity over which fly is carried on leads to a manifold increasing in the number of detected reference points, which leads to a significant costs and computational resources and as a result to the floating time of the current pair of frames processing during the entire flight. The positive side of this approach is invariance to the weather conditions, lighting, image angular rotations and region over which the flight is carried out, and drawback – large consumption of computing power, which is limited, resulting in a search for compromises using a mathematical methods.

Constructing the visual navigation system is clear a choice of the "from frame to frame" approach. The existing and widely used methods from the computer vision field for describing "descriptor" reference point and to find a pair, such as: SURF, SIFT, ORB, ACAZE has shown sufficient efficiency, but also a great computing power consumption because of its non-optimal program realization. The main drawback in this case is the lack of parallel or data-flow computing, as well as the predetermined limits of number and parameters of key points, which is redundant on the regions with good informativity and insufficient on the low one. Therefore, in this paper for solving this problem are proposed to develop and train a neural network (NN), which will allow us to search the key points and their pair on the basis of the above algorithms, but with parallel computation and texture analyzer, which will dynamically determine the informativity of the current region, and pick up each time the optimal search parameters. For realization of the NNs it was proposed to

978-1-5090-1052-3/16/$31.00 © 2016 IEEE

choose from the most powerful and affordable airborne computers microcomputer ODROID XU4 or FPGA.

IV. Convolution Neural Network as an Effective Video Processing Mean

To solve this problem as the NN is proposed to use a convolution neural network (CNN).

The CNN idea is in interlacing of Convolution layers and subsampling layers. Network structure – a one-directional (without feedback), multi-layer (Fig 1.):

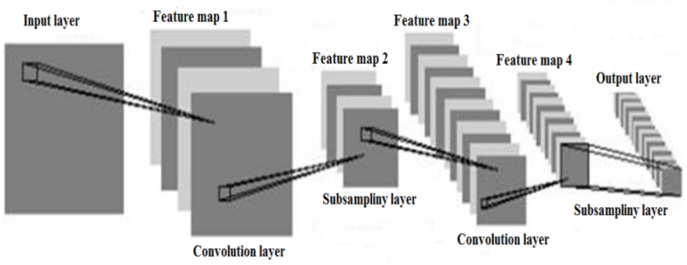

Fig. 1. Convolution network structure.

Convolution network model consist of three layers types: convolutional layers, subsampling layers and layers of a «usual» neural network – perceptron.

Convolution neural network architecture implements three ideas that provides the invariance of the network to small shifts, scaling and distortions:

– each neuron receives input signal from a local receptive field in a previous layer, providing a two-dimensional local neuronal connectivity;

– each hidden layer of the network consists of feature maps set, on which all neurons have a shared weights, which provides invariance to displacement and reduce the total number of the network weight coefficients;

– after each convolution layer follows a computing layer, which performs local averaging and subsampling that reduces resolution of feature maps.

Convolution neural network operation is provided by two main elements.

1) Filters (features detectors).

2) feature maps.

Filter is a small matrix that represents a feature, which is necessary to find on the original image. With the help of upper filter is determined the original image parts with vertical lines, the lower filter is used to define the image parts with horizontal lines.

Process of determining directly based on the operation of filter convolution the original image. The convolution results, which determine the location of the original image features is called *feature maps*.

The convolution process goal is to reduce the feature map dimension to such an extent that a feedforward network (in most cases multilayer perceptron) could operate with full features set.

Convolution layer realizes the local receptive fields idea, i.e., each output neuron is connected only to a certain (small) area of the input matrix and thus simulates some features of a human vision.

Drawbacks of CNN.

1) The high architecture complexity.
2) Fully connected.
3) Fixed area of the convolution layer window.

In order to improve the CNN operation efficiency the optimal values of the following parameters must be found:

- feature maps number;
- penetration of connections between feature maps;
- window size;
- overlapping area;
- initial weights' initialization.

V. Method of Training of Convolution Neural Network

Convolution neural network is trained by the error backpropagation

$$k_j^{l,p+1} = k_j^{l,p} - \alpha_l^k \mathrm{rot}180\left(x^{l-1}\mathrm{rot}180\left(\delta_j^l\right)\right),$$

where k_j^l is the convolution kernel of the *l*th layer of the *j*th feature map of the *p*th training stage; α_l^k is the training speed; x^{l-1} is the *l*–1 layer input; $\mathrm{rot}180\left(\delta_j^l\right)$ is the inverted error matrix for the selected kernel; δ^l is the error at the output of the convolution layer formed by simply increasing of the error matrix size of the next to it subsampling layer

$$\delta^l = upsample\left(\delta^{l+1}\right)\cdot f'\left(u^l\right),$$

where δ^{l+1} is the error of the *l*+1 layer; $f'\left(u^l\right)$ is the derivative of the activation function; u^l is the state (not activated) of the neuron layer *l*; *upsample* (•) is the order of matrix operation.

$$b_j^{l,p+1} = b_j^{l,p} - \alpha_l^b \sum \delta_j^l,$$

where b_j^l is the coefficient of the convolution layer shift for *p*th training stage; $\sum \delta_j^l$ is the shear gradient of the convolution layer *l*; α_l^b is the learning speed.

$$a_j^{l,p+1} = a_j^{l,p} - \alpha_l^a \delta_j^l subsample\left(x^{l-1}\right),$$

where $a_j^{l,p+1}$ are coefficients of the *l*th subsampling layer on p+1 training stage; x^l is the output of layer *l*; α_l^a is the training speed; *subsample* $(\bullet)$ are local maximum values sampling operation; δ^l is the error of layer *l*.

$$\delta^l = f'(u^l)\sum_{j=1}^{n}\delta^{l+1}\text{rot}180\left(k_j\right),$$

where rot180(k) is the reversed kernel; $f'(u^l)$ is the derivative of the activation function; u^l is the state (not activated) of the neuron layer *l*; *k* is the convolution kernel.

$$b_{s,j}^{l,p+1} = b_{s,j}^{l,p} - \alpha_l^{s,b}\sum\delta_j^{s,l},$$

where $b_{s,j}^{l,p+1}$ is the coefficient of the subsampling layer shift for p-th training stage; $\alpha_l^{s,b}$ is the training speed; δ^l is the error of layer *l*; $\sum\delta_j^{s,l}$ is the shear gradient of the subsampling layer *l*.

$$W^{l,p+1} = W^{l,p} - \alpha_l^w\left(\delta^l\right)^{\mathrm{T}}\cdot x^{l-1},$$

where $W^{l,p+1}$ is the weighting matrix MLP of *p*th training stage; x^{l-1} is the input of layer *l*; δ^l is the error of layer *l*; α_l^w is the training speed.

The basic element of the CNN is a classifier, which is usually perceptron or Softmax. In this paper, to improve the efficiency proposed to use the NN NEFCLASS with pretraining based on the restricted Boltzmann machine, which greatly increases the CNN efficiency.

VI. Fuzzy Classifier Pretraining with the Help of the Restricted Boltzmann Machine

NEFCLASS network belongs to the class of the three-layers indeterminate perceptrons, to which NEFCON and NEFPROX and their different modifications also belongs.

The structure of the fuzzy rules that describes the data looks like follows:

If x_1 belongs to $\mu_1,\ldots,x_n$ belongs to μ_n, then template $x_1,\ldots,x_n$ belongs to class *i*, where $\mu_1,\ldots,\mu_n$ – membership function.

System NEFCLASS has 3-layers series structure (Fig. 2).

1) First layer U contains input neurons and process input data. Activation a_x^{1} of neuron $x \in U$, doesn't changed input value.

2) Neurons of the hidden layer U_2 contain fuzzy rules. Neurons activation function is

$$a_R^{(p)} = \min_{x\in U_1}\left\{W(x,R)(a_x^{(p)})\right\},$$

where $W(x,R) = \mu_i(x)$) is the weight of the input neuron and rules layer *R* connection.

3) Third layer U_3 consists of each class input neurons. The output value is calculated as follows:

$$a_c^{(p)} = \max_{x\in U_1}\left\{a_R^{(p)}\right\}.$$

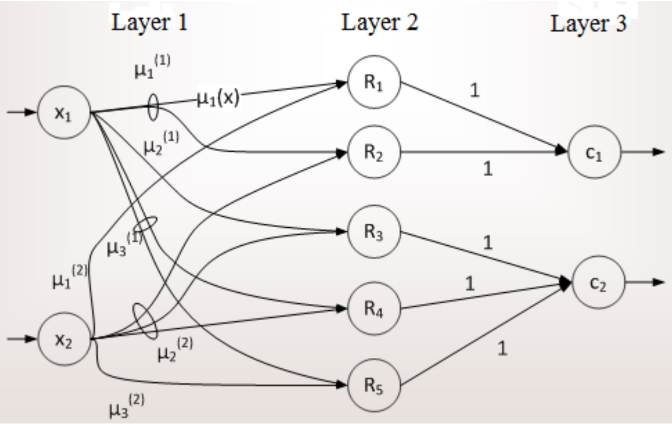

Fig. 2. Structure of the NEFCLASS network.

For pretraining activation functions were calculated on a separate neurons' layer and then was the distribution according to rules. Pretraining to be possible, the activation function must be replaced by a sigmoidal one of the following form

$$\frac{1}{1+e^{-a(x-c)}}.$$

Final network topology is shown in Fig. 3.

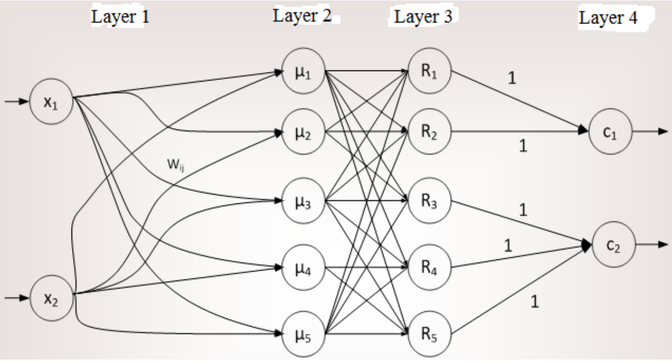

Fig. 3. Modified structure of the NEFCLASS network.

The restricted Boltzmann machine was chosen as the training network. According to the paradigm of pretraining using an autoencoders, replace the input layer and layer, which calculate the membership functions on a restricted Boltzmann machine (Figs 4 and 5).

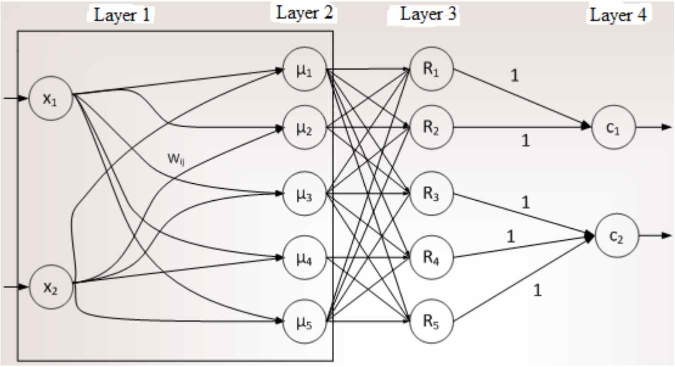

Fig. 4. Selection of the restricted Boltzmann machine.

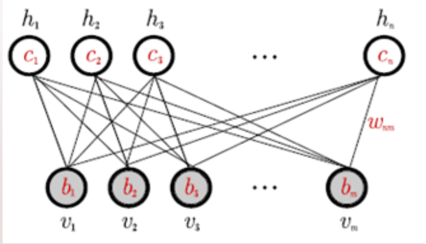

Fig. 5. Restricted Boltzmann machine structure.

VI. Conclusion

Implementation of intelligent image recognition system based on convolution neural network with deep leraning, reduce amount of requiring onboard microcomputer resources, and increase accuracy of ladnmarks recognition. Programm algorithms that based on neural network allows it's parallel processing implementation that greatly reduce time of calculation. Learing sample gives opportunity to realise stable image recognition results on different types of textures earth's surface.

References

[1] R.M. Bell and Y. Koren. Lessons from the netflix prize challenge. ACM SIGKDD Explorations Newsletter, 9(2):75–79, 2007.

[2] A. Berg, J. Deng, and L. Fei-Fei. Large scale visual recognition challenge 2010. www.imagenet.org/challenges. 2010.

[3] L. Breiman. Random forests. Machine learning, 45(1):5–32, 2001.

[4] D. Cire¸san, U. Meier, and J. Schmidhuber. Multi-column deep neural networks for image classification. Arxiv preprint arXiv:1202.2745, 2012.

[5] D.C. Cire¸san, U. Meier, J. Masci, L.M. Gambardella, and J. Schmidhuber. High-performance neural networks for visual object classification. Arxiv preprint arXiv:1102.0183, 2011.

[6] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In CVPR09, 2009.

[7] J. Deng, A. Berg, S. Satheesh, H. Su, A. Khosla, and L. Fei-Fei. ILSVRC-2012, 2012. URL http://www.image-net.org/challenges/LSVRC/2012/.

[8] L. Fei-Fei, R. Fergus, and P. Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. Computer Vision and Image Understanding, 106(1):59–70, 2007.

[9] G. Griffin, A. Holub, and P. Perona. Caltech-256 object category dataset. Technical Report 7694, California Institute of Technology, 2007. URL http://authors.library.caltech.edu/7694.

[10] G.E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R.R. Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors. arXiv preprint arXiv:1207.0580, 2012.

[11] K. Jarrett, K. Kavukcuoglu, M. A. Ranzato, and Y. LeCun. What is the best multi-stage architecture for object recognition? In International Conference on Computer Vision, pages 2146–2153. IEEE, 2009.

[12] A. Krizhevsky. Learning multiple layers of features from tiny images. Master's thesis, Department of Computer Science, University of Toronto, 2009.

[13] A. Krizhevsky. Convolutional deep belief networks on cifar-10. Unpublished manuscript, 2010.

[14] A. Krizhevsky and G.E. Hinton. Using very deep autoencoders for content-based image retrieval. In ESANN, 2011.

[15] Y. Le Cun, B. Boser, J.S. Denker, D. Henderson, R.E. Howard, W. Hubbard, L.D. Jackel, and et al. Handwritten digit recognition with a back-propagation network. In Advances in neural information processing systems, 1990.