IEEE Second International Conference on Data Stream Mining & Processing
August 21-25, 2018, Lviv, Ukraine

Model and Training Methods of Autonomous Navigation System for Compact Drones

Viacheslav Moskalenko
dept. of computer science
Sumy State University
Sumy, Ukraine
v.moskalenko@cs.sumdu.edu.ua

Alona Moskalenko
dept. of computer science
Sumy State University
Sumy, Ukraine
a.moskalenko@cs.sumdu.edu.ua

Artem Korobov
dept. of computer science
Sumy State University
Sumy, Ukraine
a.korobov@cs.sumdu.edu.ua

Olha Boiko
dept. of computer science
Sumy State University
Sumy, Ukraine
o.shulyma@ssu.edu.ua

Serhii Martynenko
dept. of computer science
Sumy State University
Sumy, Ukraine
serg.martynenko@gmail.com

Oleksandr Borovenskyi
dept. of computer science
Sumy State University
Sumy, Ukraine
aleks.borovensky@gmail.com

***Abstract* — The paper presents a novel model of convolutional neural network for visual feature extraction, support vector machine for position prediction and information-extreme classifier for obstacle prediction with new training methods to build decision rules of autonomous navigation system for compact drones are presented in the paper. Sparse-coding neural gas algorithm for unsupervised training of the convolution filters, supervised incremental learning method for training the regression model and particle swarm optimization algorithm for training the classifier model are proposed. The complex criterion for choosing parameter of feature extractor model is considered. Simulation results with optimal model on test open datasets confirm the suitability of proposed algorithms for practical usage.**

Keywords—navigation, visual odometry, convolutional neural network, neural gas, information criterion, support vector regression

I. Introduction

Unmanned aerial vehicles (UAV) are widely used in precision agriculture, search and rescue operations, transport and aerial filming. Development information technology which lowers the demands on UAV hardware resources and improves the reliability of autonomous decision-making under constantly changing environmental conditions and variability of objects of interest allows to reduce the system's weight and cost whilst simultaneously expanding the functionality of the onboard system. One of the ways to improve the functional efficiency of the UAV system is to use machine vision and machine learning to build data analysis models based on visual and inertial sensor data [1-3].

The use of functional navigation systems based on the comparison of visual features requires the availability of a database of reference images, This makes is potentially unsuitable for situations requiring a rapid response to changes in the environment [4]. Visual odometry and Simultaneous Localization And Mapping (SLAM) methods are less efficient in poorly textured scenes and in the presence of non-static elements in the field of view [5, 6]. Moreover, the deployment of these technologies requires significant computational resources, which limits their use in autonomous compact UAVs.

Today, the convolutional neural network, consisting of a multilayer feature extractor based on convolution filters and decisive rules in the form of fully connected neural layers, is an undisputed leader among the image analysis models [7, 8]. However, the essential disadvantages of traditional convolutional neural networks lay in their inability to analyze the processes occurring in time, as well as the high computational complexity of the backpropagation-based learning algorithm, which makes adapting to changes in operating conditions difficult. Conversely, using unsupervised learning methods based on sparse-coding neural gas to train neural networks shows promise. It reduces both the required quantity of labeled observations and computational load [9].

This paper proposes a model of convolutional neural network for analysis of spatial-temporal patterns to be used in autonomous navigation and identification of obstacles under computational resource constraint. We also propose a training method for such network based on unsupervised learning combined with decision rules based on support vector machines [10] and intellectual information-extreme technology [11]. The results of parameter optimization and testing of proposed algorithms on real-life open source data sets are considered.

II. Materials and methods

Let an annotated set of video frames be formed $\{ c_t =< v_t, x_t, y_t, z_t, a_t > | t = \overline{1,n} \}$, where v_t – frame image at time *t* and x_t, y_t, z_t – camera coordinates obtained from the Global Positioning System (GPS) and converted to the North East Down (NED) local coordinate system, $a_t \in \{A_r^o \mid r = \overline{1,R}\}$ – operator's response to the obstacle, where A_r^o denotes a recognition class that characterizes the obstacle.

A structured vector of space-time parameters of the UAV navigation system operationin general has a structure :

$$g =< e_1,...,e_{\xi_1},...,e_{\Xi_1}, f_1,..., f_{\xi_2},..., f_{\Xi_2} >, \ \Xi_1 + \Xi_2 = \Xi, \quad (1)$$

Lviv Polytechnic National University Institutional Repository http://ena.lp.edu.ua

where $< e_1,...,e_{\xi_1},...,e_{\Xi_1} >$ – genotype model parameters which affect the parameters of the feature extraction algorithms; $< f_1,..., f_{\xi_3},..., f_{\Xi_3} >$ – phenotypic model parameters which influence the decision rules.

At the same time, known limitations on the corresponding model parameters are:

$$R_{\xi_1}(e_1,...,e_{\xi_1},...,e_{\Xi_1}) \le 0 ; \quad R_{\xi_2}(f_1,..., f_{\xi_2},..., f_{\Xi_2}) \le 0 .$$

The process of machine learning of the navigation system is focused on determining the optimal coordinate values of the vector (1), which provide the maximum of the complex criterion

$$J = \frac{\overline{E}}{E_{\max}} \cdot \frac{\varepsilon_{\min}}{\varepsilon} \cdot \frac{C_{\min}}{C}, \quad (2)$$

$$g^* = \arg\max_G \{J(g)\} , \quad (3)$$

where $\overline{E}$ –information criterion of learning effectiveness for the recognition of obstacle averaged by the set of classes; ε – the value of the mean square error of regression when determining the change of camera coordinates in space; C – the criterion of computational complexity of feature extraction algorithms; $E_{\max}$, $\varepsilon_{\min}$, $C_{\min}$ – the maximum possible value of the informational criterion of classifier training efficiency and the minimum allowable values of regression model error and the criterion of computational complexity of the system's algorithms, respectively.

For the formation of the input mathematical description of the intelligent information system, KITTI Vision Dataset [8] training kits a containing both the frame sequence of the image from the moving video camera and the movement data along three coordinate axis reported by GPS and LiDaR [8]. To train the model, movement data is converted to the local NED coordinate system and the relative movements of the camera $\Delta x, \Delta y, \Delta z$ are determined between adjacent video frames.

The schematic of the intelligent navigational system for a compact UAV is shown in Fig. 1. In order to extract the feature representation of visual observations, it is proposed to use a convolutional neural network, using a multichannel image formed by a series sampling of successive video footage in grayscale format as input. The convolutional neural network has a multilayered structure to form a high-level feature representation of observation results, with convolutional filters trained in unsupervised manner successively layer by layer. An information-extreme classifier trained in supervised mode on the training samples encoded by the corresponding high-level features is used for obstacle prediction and output of the corresponding reaction. The regression model in the framework of the support vector machine is used to map the visual features and the data from inertial sensors into the corresponding estimation of the displacement of the video camera in space.

Fig. 2 shows the 4-layer architecture of the convolutional neural network, in the first layer of which there are 3D-filters of different scales: 5x5x K_1 , 3x3x K_1 and 1x1x K_1 . The number of filters is regulated by the parameter K_2 . To preserve the same size of character maps formed by multiple-scale filters, the technique of padding with zeros is used [8]. In the second and third layers, stride parameter of scanning a feature map with multiple-scale filters is 3 and 2, respectively.

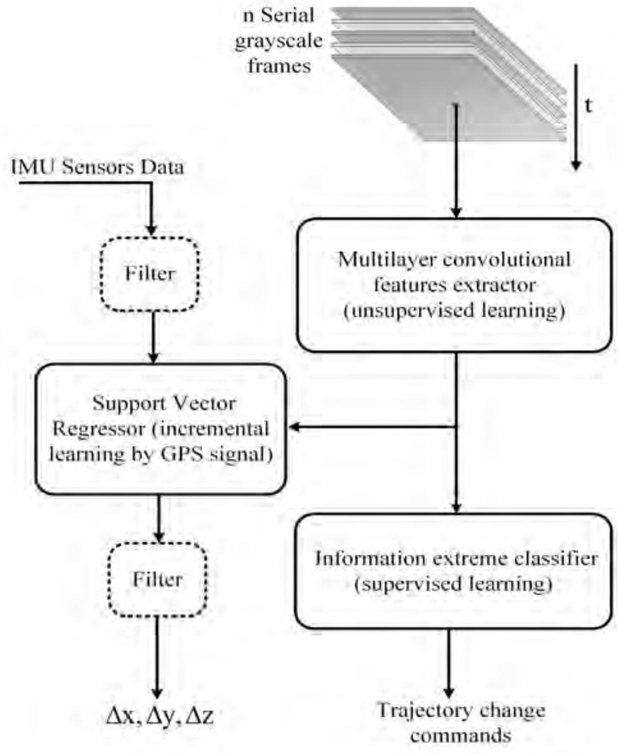


Fig. 1. A generalized scheme of intelligent navigational system of small size UAV

Fig. 2 does not show the activation function applied to each feature map. We propose to use the Orthogonal Matching Pursuit algorithm [9] for calculating a response on each feature map and rectifier function $y = \max(0, x)$, however, to avoid information loss, we can double the feature map using the following function: $y = \{\max(0, x), \max(0, -x)\}$.

An important step of data analysis is a preliminary normalization with the view to removing linear correlation of components of observation and the unification of primary feature representation. Data whitening with the use of the method of ZCA (Zero-phase Component Analysis) is one of the most common methods of preliminary data normalization. ZCA method implies performance of the following steps:

1) calculation of mean selected value of features μ = mean(X);

2) calculation of co-variative matrix of selected observations Σ: = cov(X);

3) singular decomposition of co-variative matrix $\Sigma \approx$ VDTT;

Lviv Polytechnic National University Institutional Repository http://ena.lp.edu.ua

4) whitening of each observation by formula $x_j := VD^{-1/2}V^T(x_j - \mu)$.

Unsupervised learning of convolutional filters is proposed to be carried out in accordance with the algorithm of sparse-coding neural gas, which was considered and studied in [9]. The input data for the algorithm of sparse-coding neural gas is the power of the set of the basis vectors M, the dimension of feature space N, $\lambda_0, \lambda_{final}$ – the initial and final value of the neighborhood size, η_0, η_{final} – the initial and final values of the learning rate.

Consider the main steps of the algorithm.

1) Initialization of the dictionary of basis vectors $D = (d_1, ..., d_M)$ by random numbers with uniform distribution;

2) Initialization of the counter of training vectors $t := 1$.

3) Choosing a random vector x from the set of training vectors X.

4) The normalization of vectors from the dictionary $D = (d_1, ..., d_M)$ by bringing it to a unit length.

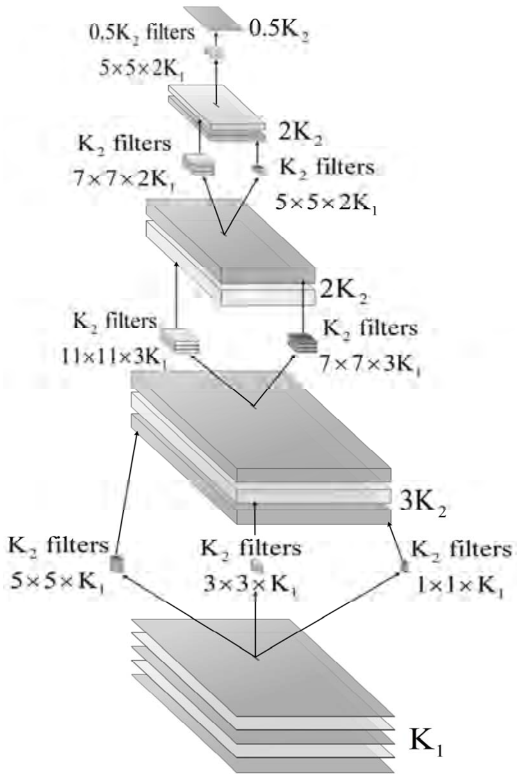


Fig. 2. The architecture of the convolutional neural network for visual feature extraction in the UAVs navigation system

5) Calculation of the current values of the neighborhood size λ_t and learning rate η_t:

$$\lambda_t := \lambda_0 (\lambda_{final} / \lambda_0)^{t/t_{\max}};$$

$$\eta_t := \eta_0 (\eta_{final} / \eta_0)^{t/t_{\max}}.$$

6) The similarity calculation of the input vector x to the basis vectors $d_{l_k} \in D$ for their sorting

$$-(d_{l_0}^T x)^2 \le ... \le -(d_{l_k}^T x)^2 \le ... \le -(d_{l_{M-1}}^T x)^2$$

7) Update the coordinates of the main vectors $d_{l_k} \in D$ according to the Oja's learning rule [9]

$$d_{l_k} := d_{l_k} + \eta_t \exp(-k/\lambda_t) y (x - y d_{l_k}), \ y := c_{l_k}^T x,$$
$$k = \overline{0, M-1}.$$

8) If $t < t_{\max}$, then the increment of the counter $t := t + 1$ and go to the step 3.

The information-extreme classifier for evaluation of the obstacle performs the adaptive discretization of the feature representation of dataset $\{x_{r,i}^{(j)} \mid i = \overline{1,N};\ j = \overline{1,n_r};\ r = \overline{1,R}\}$ on the basis of the coarse binary coding algorithm. This involves comparing the value of the i-th feature with the corresponding lower $T_{L,l,i}$ and upper $T_{U,l,i}$ thresholds of the asymmetric receptive field l, which are calculated by the formulas

$$T_{L,l,i} = x_{i,\max}\left[1 - \frac{\delta_{i,l}}{\delta_{\max}}\right],\ T_{U,l,i} = x_{i,\max},\ l = \overline{1,L}$$

The formation of a binary training set $\{b_{r,i}^{(j)} \mid i = \overline{1, N*L};\ j = \overline{1,n_r};\ r = \overline{1,R}\}$ is carried out according to the rule

$$b_{r,\,l*N+i}^{(j)} = \begin{cases} 1, & \text{if} \quad T_{L,l,i} \le x_{r,i}^{(j)} \le T_{U,l,i}; \\ 0, & \text{else.} \end{cases}$$

The calculation of the values of the coordinates of the binary support vector x_m, relative to which container classes are constructed on a radial basis, is carried out according to the rule

$$b_{r,l\cdot N+i} = \begin{cases} 1, & \text{if} \quad \dfrac{1}{n_r}\sum_{j=1}^{n_r} b_{r,l\cdot N+i}^{(j)} > \dfrac{1}{n}\sum_{r=1}^{R}\sum_{j=1}^{n_r} b_{r,l\cdot N+i}^{(j)}; \\ 0, & \text{else.} \end{cases}$$

Normalized modification of S. Kullback's information measure is used as a criterion of our classifier's machine learning efficiency [11]:

$$E_r = \frac{1 - (\alpha_r + \beta_r)}{\log_2(2+\varsigma) + r\log_2 10} \cdot \log_2\left[\frac{2 - (\alpha_r + \beta_r) + \varsigma}{(\alpha_r + \beta_r) + \varsigma}\right], \quad (4)$$

where α_r, β_r – false-positive and false-negative rates of classification decisions regarding the affiliation of the input

Lviv Polytechnic National University Institutional Repository http://ena.lp.edu.ua

vectors to the class A_r^o; ς – any small positive number entered to avoid uncertainty when dividing by zero.

The complexity of information-extreme machine learning increases faster than the square of the number of training vectors. Therefore, a reduction of multi-class classifier to a series of two-class classifiers is used to speed up training. The classifiers are constructed on the principle of "one-against-one", with a total of $\mathrm{M}\cdot(\mathrm{M}-1)/2$ two-class classifiers constructed [11].

In the exam mode, the decision on the affiliation of the observation x to one of the classes of set $\{A_r^o\}$ is taken according to the geometric membership function [11]

$$\mu_r^*(x)=\max_{\{r\}}\{\mu_r(x)\},$$

where $\mu_k(x)$ is the membership function of vector x to the container of class $\{A_r^o\}$ which is calculated by the rule:

$$\mu_r(x)=\exp\left(-\frac{\sum_{i=1}^{N\cdot L}(x_{r,i}^*\oplus x_i)}{radius_r^*}\right),$$

where $radius_r^*$ is the optimal radius of class container A_r^o.

To train the regression model $y=f(x)$, output variable $y^{(t)}\in R$ of which corresponds to the change of the coordinates of the camera $\Delta x,\Delta y$, or Δz, a set of $(x^{(t)},y^{(t)})_{t=1}^n$ training data, consisting of visual features and measurements of inertial sensors, is used, where $x^{(t)}\in R^N$. The regression function is linear in the secondary feature space and has the following form

$$f(x)=(\omega,\varphi(x))+b, \tag{5}$$

$$\varphi: R^{n}\to H,\ \omega\in H. \tag{6}$$

where ω and b are empirical coefficients which can be obtained through training; H – multidimensional space of secondary features.

The coefficients ω and b can be found by minimizing the following formula:

$$\min R(\omega,\xi,\xi^*)=\frac{1}{2}\|\omega\|^2+\Psi\sum_{t=1}^{n}(\xi_t^*+\xi_t)$$

$$\begin{aligned}&y^{(t)}-(\omega,\ \varphi(x))-b\le\varepsilon+\xi_t^*\\&(\omega,\ \varphi(x))+b-y^{(t)}\le\varepsilon+\xi_t\\&\xi_t^*,\xi_t\ \ge 0,\ i=1,2,\dots,n,\ \varepsilon\ge 0.\end{aligned} \tag{7}$$

where Ψ – coefficient of regularization; ξ, ξ^* – slack variable, which measures the measurement uncertainty from below and above, respectively; ε – the insensitivity of the loss function, which means that if $f(x)$ is in the range $y^{(t)}\pm\varepsilon$, then the measurement uncertainty is not taken into account.

The optimization problem (7) is a quadratic programming problem with linear constraints, which can be solved by introducing Lagrange multipliers and applying the Karush-Kuhn-Tucker conditions to solve a dual problem. [10]:

$$\begin{aligned}\min R(\nu,\nu^*)=&\sum_{t=1}^{n}(\nu_t^*-\nu_t)(\nu_j^*-\nu_j)K(x^{(t)},x^{(j)})+\\&+\varepsilon\sum_{t=1}^{n}(\nu_t^*+\nu_t)-\sum_{t=1}^{n}y^{(t)}(\nu_t^*-\nu_t)\\&\sum_{t=1}^{n}(v_t^*+v_t)=0\end{aligned}$$

$$0\le v_t,\ v_t\ \le\frac{\Psi}{l},\ t=1,2,\dots,n \tag{8}$$

where v_t and v_t^* are Lagrange multipliers associated with constraints (8);

$$K(x^{(t)},x^{(j)})=\varphi(x^{(t)})\cdot\varphi(x^{(j)}).$$

A typical example of a kernel function is a polynomial kernel and a Gaussian kernel. In general, the regression function has the form

$$f(x)=\sum_{t=1}^{n}(v_t^*-v_t)K(x^{(t)},x)+b$$

Not all training samples can become support when support vectors are used for training the regression model. Only vectors on the boundary have the probability of becoming support. The incremental training of a regression model on support vectors can be realized by determining the convex border of discrete points when choosing a set of boundary vectors as a set of training ones. In this case, the convex border of discrete points is the border, which can surround all the discrete points, formed by the outermost point through connections. Therefore, after processing of the first sub-sample, the formed support vectors are compared with the vectors of the following subclasses at an angle of inclination, to form a plurality of boundary vectors, as vectors of the maximum inclination. Each step of the supplement may be accompanied by retraining.

(7)

III. SIMULATION RESULTS AND DISCUSSION

To train the feature extractor, both the training and test video sequences of the KITTI Vision Dataset set are used, without taking annotations into account. To reduce the computational complexity of the algorithms, the images are compressed to a resolution of 200x200 pixels. In this case, the procedure is repeated for different values of parameters K_1 and K_2, which affect both the informative nature of the

Lviv Polytechnic National University Institutional Repository http://ena.lp.edu.ua

feature representation and the computational complexity. We propose to measure complexity by the quantity of Mul and Add operations performed during the convolutional operations with an image or a feature map. For the network architecture shown in Fig. 2, the complexity can be calculated as

$$C = K_2(2706472K_1 + 4438784K_2)\ . \qquad (9)$$

For the classifier and regression model, the optimal configuration of the convolutional extractor may be different as they are responsible for different tasks. Therefore, a complex criterion (2) offers a compromise from the point of view of the accuracy of the decision rules and the computational complexity of the extractor of visual features.

For our support vectors of our regression model, we propose to use a Radial basis kernel in the following form

$$K(x^{(t)}, x^{(j)}) = \exp(\gamma \left\| x^{(t)} - x^{(j)} \right\|^2),\ \gamma \ge 0$$

where γ – the kernel coefficient, the default value of which is $\gamma = 1/N$.

The set of recognition classes $\{A_r^o\}$ is describing the characteristic obstacles and the corresponding reaction commands, and has a power $R = 5$. The first class of recognition A_1^o characterizes the normal state of following a prescribed trajectory. The classes A_2^o and A_3^o correspond to the left turn of 45 and 90 degrees respectively. The classes A_4^o and A_5^o correspond to the right turn of 45 and 90 degrees respectively. The volume of the training samples of each class is $n_r = 500$.

The optimization of the parameters of the receptive field $\{\delta_{m,l,i}\}$ and other genotype parameters for the information-extreme classifier amounts to finding the extremum of the criterion function (4) in the hyperspace of solutions. For the purposes of this it is suggested to use a Particle Swarm Optimization algorithm (PSO) [11]. The effectiveness of each particle of a population algorithm, which lies in its proximity to the global optimum, is measured by means of a predetermined fitness function. This role is fulfilled in our case by the training efficiency criterion (4). In this case, the following parameters of the population algorithm configuration are specified : maximum particle speed $V_{\max,i}$ = 2, particles acceleration constants $c_1 = c_2 = 1$, the number of swarm agents $n_a = 100$, the coefficient of inertia $w = 0,95$ and the number of iterations $K_{ITER} = 3000$.

The optimization of the phenotypic parameters of the decision rules (radii of container classes) can be carried out by the direct search with a given step, since the number of steps for such a search is relatively small. To identify the tendency of changing in average values of the partial and complex criteria during the growth of parameters K_1 and K_2, which affect the size of the convolution extractor (Fig. 2), a simulation was performed for the three fixed values of each of these parameters (Table I).

TABLE I. DEPENDENCE OF PARTIAL AND COMPLEX CRITERIA ON EXTRACTOR PARAMETERS OF A FEATURES DESCRIPTION K_1 AND K_2

K_1	K_2	$\bar{E} / E_{\max}$	$\varepsilon_{\min} / \varepsilon$	$C_{\min}$ /C	J
3	4	0,083	0,112	1,000	0,009296
5	4	0,101	0,188	0,827	0,015703
7	4	0,098	0,200	0,705	0,013818
3	8	0,28	0,688	0,297	0,057214
5	8	0,29	0,756	0,264	0,057879
7	8	0,29	0,775	0,238	0,053491
3	16	0,39	0,968	0,082	0,030957
5	16	0,55	1,000	0,077	0,04235
7	16	0,51	1,000	0,072	0,03672

The analysis of table 1 shows that an increase in parameter values K_1 and K_2 in general leads to an increase in the reliability and computational complexity (9) of the decision rules of the classifier and the regression model. At the same time, the increase of the parameter K_1 has little effect on the efficiency of the classifier due to the decrease in the efficiency of the swarm search with a significant increase in the size of the feature space, while the regression error is equally sensitive to the value of parameters K_1 and K_2.

However, given that with growth in K_1 and K_2 the reliability of decision rules grows more slowly than the computational complexity use of complex criterion J offers a suitable compromise. That is, we consider the following parameter values to be optimal $K_1^* = 5$ and $K_2^* = 8$.

In the optimal configuration of the feature extractor, the average value of the information criterion of functional efficiency is $\bar{E} = 0,29$. This corresponds to accuracy of 95,2% for the training set, and 94% for the test dataset. The number of receptive fields per primary feature is L =3, chosen as the minimum value at which the information criterion (4) ceases to grow on the test dataset. Fig. 3 shows a graph of the change of the average information efficiency criterion (4) in relation to the number of iterations of the particle swarm search algorithm.

The analysis of Fig. 3 shows that after a 1000th iteration growth of the information criterion (4) has begun to slow down, and after 2500th iteration remained virtually unchanged. Such a change in the criterion indicates that the further increase in the information criterion is achievable only with the increase in the informative nature of the features by increasing values of K_1 and K_2 or improving the structure of the extractor (Fig. 2).

For a visual assessment of the effectiveness of the machine learning of the navigation system, a reference trajectory measured using GPS and LiDaR can be compared with a reconstructed trajectory obtained using a trained model.

Fig. 4a shows the reference trajectory (dashed line) and the reconstructed trajectory (solid line) created by the proposed algorithms on the basis of the test data from the KITTI database [8]. Fig. 4b shows the results of a similar experiment, but using the model proposed in [8].

The analysis of Fig. 4 shows that the accuracy of reconstruction of the trajectory in both cases is acceptable for practical use and does not differ significantly. Notably, however, the proposed model has much fewer parameters and allows the use of an unsupervised training instead of a computationally intensive gradient descent algorithm.

Lviv Polytechnic National University Institutional Repository http://ena.lp.edu.ua

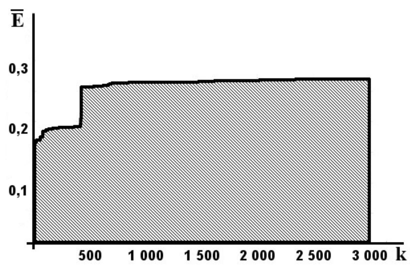
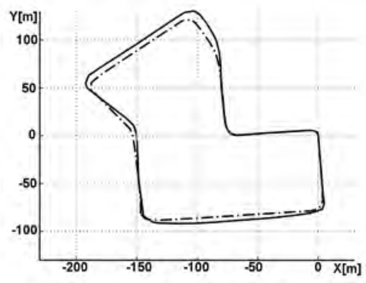
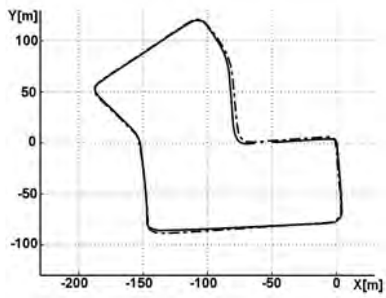


Fig. 3. A graph of the change of the average information efficiency criterion (4) in dependence from the number of iterations of the optimization swarm search algorithm



a)



b)

Fig. 4. Reference and reconstructed trajectory: a – the developed model; b – the model proposed in the work [9]

IV. Conclusion

1. The scientific novelty of the results is as follows:

– a new model for the autonomous navigation system of a compact UAV is proposed for the first time. The constituent parts of the model are a feature extractor trained without supervision, a support vector regression model, which can be incrementally trained under supervision on the visual and inertial sensor data, and an information-extreme obstacle classifier, which learns to react to obstacles under supervision, which in turn reduces the computational resource requirements;

– a model of a 4-layer convolutional network using as inputs a series of successive frames which are interpreted as channels of one image and scanned by multiple-scale filters is proposed for the first time;

– a method of unsupervised training of the convolution filters based on sparse-coding neural gas, which allows training simultaneous with direct propagation of the signal without using the back error propagation is proposed for the first time;

– a method for evaluating the effectiveness of the data analysis model in navigation problems was improved with the application of multiplicative convolution of partial criteria. This allows to select the optimal system parameters in the information and computational cost sense.

2. The practical value of the obtained results for unmanned aviation lies in the formation of a modern scientific and methodological basis for designing compact autonomous navigation systems for UAVs operating under resource and information constraints, and capable of learning. At the same time, the results of the simulation model confirm the high efficiency of the resulting decision rules for determining the coordinates in space and recognition of obstacles based on the video stream and inertial sensor data.

Acknowledgment

The work was performed in the laboratory of intellectual systems of the computer science department at Sumy State University with the financial support of the Ministry of Education and Science of Ukraine in the framework of state budget scientific and research work of DR No. 0117U003934.

References

[1] S. Wang, Z. Deng, and G. Yin, An Accurate GPS-IMU/DR Data Fusion Method for Driverless Car Based on a Set of Predictive Models and Grid Constraints,’ Basel, Switzerland : Multidisciplinary Digital Publishing Institute, vol. 16(3), pp. 280–293, 2016.

[2] B. Suwandi, T. Kitasuka, and M. Aritsugi, “Low-cost IMU and GPS fusion strategy for apron vehicle positioning,” TENCON 2017, IEEE Region 10 Conference, Penang, Malaysia, pp. 449–454, Nov. 2017.

[3] B. A. Mary, and P. H. Gerhard, “Pose Estimation of a Mobile Robot Based on Fusion of IMU Data and Vision Data Using an Extended Kalman Filter,” Basel, Switzerland : Multidisciplinary Digital Publishing Institute, vol. 17(10), pp. 2164, 2017.

[4] J. Folkesson, J. Leederkerken, R. Williams, and A. Patrikalakis, “A Feature Based Navigation System for an Autonomous Underwater Robot,” In: Laugier C., Siegwart R. (eds) Field and Service Robotics. Springer Tracts in Advanced Robotics. Springer : Berlin/Heidelberg, 2008. vol. 42, pp. 105-114.

[5] C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. Reid, and J. J. Leonard, “Past, Present, and Future of Simultaneous Localization And Mapping: Towards the Robust-Perception Age,” IEEE Transactions on Robotics, vol. 32, no. 6, pp. 1309–1332, 2016.

[6] G.-L. Dorian, S. Marta, D. T. Juan, and J.M.M. Montiel, “Real-time monocular object SLAM. Robotics and Autonomous Systems,” North-Holland Publishing Co. : Amsterdam, Netherlands, 2016. vol.75, i. PB, pp. 435–449.

[7] Th. Ayoul, T. Buckley, and F. Crevier. UAV Navigation above Roads Using Convolutional Neural Networks Available from: http://cs231n.stanford.edu/reports/ 2017/pdfs/553.pdf.

[8] V. Mohanty DeepVO: A Deep Learning approach for Monocular Visual Odometry. Computer Vision and Pattern Recognition, 2016. Available from: https://arxiv.org/pdf/1611.06069.pdf

[9] K. Labusch, E. Barth, and T. Martinetz, “Sparse Coding Neural Gas: Learning of Overcomplete Data Representations,” Neurocomputing, Elsevier Science Publishers B. V.: Amsterdam, Netherlands, vol. 72, is. 7–9, pp. 1547–1555, 2009.

[10] H. Xu, R. Wang, and K. Wang. A New SVR Incremental Algorithm Based on Boundary Vector. 2010. Available from: https://www.researchgate.net/publication/238519783_A_New_SVR_Incremental_Algorithm_Based_on_Boundary_Vector

[11] V. V. Moskalenko, and A. G. Korobov, “Information-extreme algorithm of the system for recognition of objects on the terrain with optimization parameter feature extraction,” Radio Electronics, Computer Science, Control. Zaporizhzhya National Technical University : Zaporizhzhya, Ukraine, no. 2, pp. 38–45, 2017.

Lviv Polytechnic National University Institutional Repository http://ena.lp.edu.ua