COMPUTER SCIENCES AND INFORMATION TECHNOLOGIES

UDC 004.032.26(045)
DOI:10.18372/1990-5548.80.18678

[1]**V. M. Sineglazov,**
[2]**D. O. Nikulin**

INTELLIGENT SYSTEM OF GENERATION OF CAMOUFLAGE PATTERNS BASED ON ARTIFICIAL INTELLIGENCE TECHNOLOGIES

[1,2]Aviation Computer-Integrated Complexes Department, Faculty of Air Navigation Electronics and Telecommunications, National Aviation University, Kyiv, Ukraine
E-mails: [1]svm@nau.edu.ua ORCID 0000-0002-3297-9060,
[2]dmytronikulin9@gmail.com

Abstract—*The work is devoted to the development of an intelligent system for generating camouflage patterns based on artificial intelligence technologies. A generative-competitive network is used as an intellectual element of this system. To solve the problem of the collapse mode, the architecture of progressively growing GANs (ProGAN) is used. The system allows you to generate completely new camouflage patterns for the selected area by iteratively improving the pattern. Due to the mechanism of restrictions, it is possible to fix the desired aspects of the drawing (color scheme, pattern, number of colors) from an existing drawing and adapt it to the desired area. The system provides the possibility of generating micropatterns on the drawings to improve camouflage at close distances. When evaluating a camouflage pattern, the system takes into account additional parameters, such as angle (from the ground and air), time and weather.*

Index Terms—Artificial neural networks; artificial Intelligence; intelligent generation system; generative-competitive network; progressively growing GANs; camouflage patterns.

I. INTRODUCTION

The military uniform is an integral part of the equipment of military personnel, performing important functions: camouflage, protection and identification.

Traditional camouflage colors are usually based on a combination of different colors and patterns imitating the natural environment. However, with the development of technologies such as thermal imaging and infrared sensors, these camouflage patterns may not be effective enough. Therefore, there is a need to create more advanced camouflage systems that would provide protection not only in the visible range of the spectrum, but also in infrared and other ranges.

One of the promising directions in solving this problem is the use of artificial intelligence. The rapid development of artificial intelligence technologies, in particular, deep learning methods, opens up new opportunities for automated camouflage design. Generative Adversarial Networks (GANs) are one of the promising approaches capable of generating realistic and diverse camouflage patterns adapted to different environmental conditions.

II. ANALYSIS OF MODERN APPROACHES TO MILITARY CAMOUFLAGE DESIGN AND THEIR MAIN DISADVANTAGES

Traditional methods of designing camouflage patterns are usually based on the intuition and experience of designers. They usually consist of manually creating, modifying and selecting static camouflage patterns (Fig. 1), which, according to the developers, best mask objects on certain backgrounds.

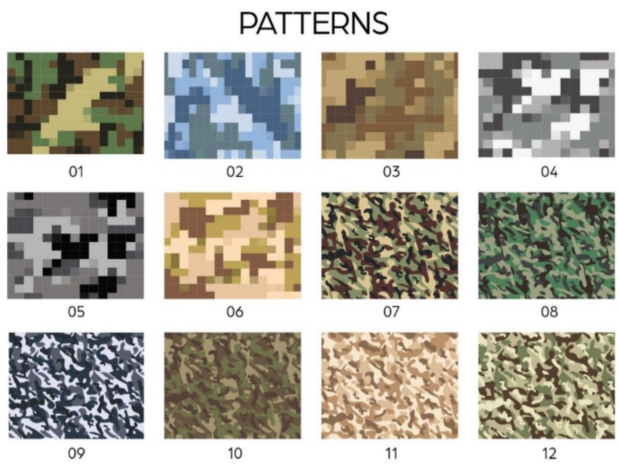

Fig. 1. An example of camouflage patterns

The main disadvantages of such approaches are:

• *Limited versatility*: Static camouflage patterns are effective only in specific environmental conditions, but do not adapt to changes in landscape, lighting, or season.

• *Suboptimality*: Manual selection of camouflage patterns without using formalized optimization methods, as a rule, does not provide maximum masking efficiency.

• *High development costs*: The creation of each new camouflage pattern requires considerable designer effort and large time and financial resources.

Traditional camouflage design methods are often limited by human imagination and experience. A GAN, on the other hand, can analyze huge amounts of data and find hidden patterns that a human might not notice. This allows for the creation of camouflage patterns that are more effective and adaptive than those created by traditional methods.

For example, a GAN can take into account the features of a particular landscape (colors, textures, lighting) and create patterns that blend perfectly with it. It can also take into account the type of object to be camouflaged (tank, plane, soldier) and create patterns that hide its shape and contours as much as possible.

Thus, GAN opens up new opportunities for designing military camouflage, allowing for more effective and innovative solutions that can significantly increase the survivability and success of military operations.

III. GENERATIVE-ADVERSARIAL NETWORK

A generative adversarial network (GAN) is a deep learning architecture. It trains two neural networks to compete with each other and generate increasingly realistic new data from a given training set (Fig. 2).

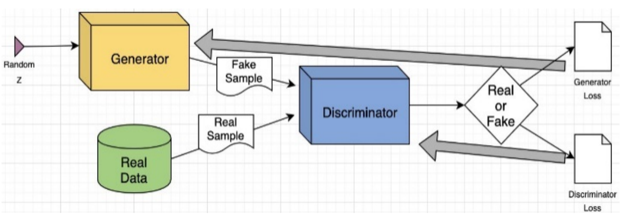

Fig. 2. GAN block diagram

The GAN training method proposed by Goodfellow is based on the idea of a competition between a generator and a discriminator. This process can be divided into several stages [1], [2]:

1) Initialization: The generator and discriminator are initialized with random weights.

2) Training the discriminator: The discriminator receives both real data from the training set and data generated by the generator. His task is to learn to distinguish between these two groups of data. For this, a loss function is used, which penalizes the discriminator for misclassifying the data.

3) Learning the generator: The generator takes random noise as input and generates new data. These data are passed to the discriminator, which evaluates their reality. The task of the generator is to learn to generate data that the discriminator cannot distinguish from real data. For this, a loss function is used, which penalizes the generator for the fact that the discriminator was able to detect a fake.

4) Repetition: Steps 2 and 3 are repeated many times. With each iteration, the generator and discriminator get better at their jobs. The generator creates more and more realistic data, and the discriminator becomes more and more demanding about the quality of the fakes.

This competition process continues until equilibrium is reached, when the generator learns to generate data that the discriminator cannot distinguish from the real thing. At this point, it can be assumed that the GAN has learned to generate data that matches the distribution of the real data.

A. Generator

The generator (G) in generative adversarial networks (GANs) is a neural network responsible for generating new, realistic data such as images. He acts as a kind of artist who learns from examples of real images and then creates new ones similar to them. In the context of camouflage generation, the generator aims to create patterns that will be as effective as possible for camouflage in various environments.

The architecture of the generator can be varied, but the following types are most often used [3]:

- *Fully Connected Networks:* This is the simplest type of architecture, where each neuron of one layer is connected to each neuron of the next layer. Such networks are good for generating simple images, but not very effective for complex camouflage patterns.
- *Convolutional Neural Networks (CNN):* CNNs are specially designed to work with images. They use convolutional layers to detect local image features such as edges, textures, and shapes. This makes them an ideal tool for generating complex camouflage patterns.
- *Deconvolutional Networks:* These networks are the inverse of CNN. They are used to increase the image size and add details. In GAN generators, they are often used to generate high-resolution images.

- *Variational Autoencoders (VAE):* A VAE is a type of generative model that learns to compress data to a smaller size (latent space) and then reconstructs it from that compressed representation. In GANs, VAEs can be used to create more diverse and controlled images.

The generator works according to the following principles [1], [4]:

1) Input data: the generator receives as input random noise (a vector of numbers) or other data that can be used to control the generation process (for example, the type of landscape, the color of the object, etc.).

2) Data processing: The generator processes the input data using its layers of neurons. Each layer performs certain operations on the data, such as convolution, activation, normalization, etc.

3) Image generation: At the last layer, the generator creates an image that is the result of processing the input data. This image should be as similar as possible to the real image from the training set.

A generator's loss function measures how well the generator performs its task. It evaluates how similar the generated image is to a real image from the training set, as well as how well it fools the discriminator. The smaller the value of the loss function, the better the generator performs its task.

The generator learns by backpropagation. This means that after each training iteration, the weights of the generator neurons are adjusted in such a way as to reduce the value of the loss function. This process is repeated until the generator learns to create images that are as close as possible to the real ones and that the discriminator cannot distinguish from the real ones.

Choosing the optimal generator architecture depends on the specific task. Fully connected networks can be used to generate simple images. For generating complex images such as camouflage patterns, it is better to use CNN or VAE.

As a conclusion, we can say that the generator is the key component of GAN, responsible for generating new, realistic data. Its architecture and operation are critical to the quality of the data generated. Choosing the optimal architecture and setting the parameters of the generator is a complex task that requires a deep understanding of the principles of GAN operation and conducting experiments with different configurations.

B. Discriminator

The discriminator (D) in generative adversarial networks (GANs) is a neural network that acts as an expert evaluator, distinguishing real images (from the training set) from fakes generated by the generator. In the context of camouflage generation, the discriminator evaluates how convincingly the generated pattern mimics real-world textures and colors, and how effectively it camouflages an object in a given environment.

Discriminator architectures can vary, including fully connected networks, convolutional neural networks (CNNs), and residual connection networks (ResNets). Fully connected networks are suitable for simple classification tasks, but are not always optimal for analyzing complex images. CNNs are the most common choice for a discriminator in GANs because they effectively detect local image features such as edges, textures, and shapes, which is important for camouflage analysis. ResNet, as a type of CNN, allows training very deep networks without loss of accuracy, which can be useful for learning complex image features.

Regardless of the chosen architecture, the discriminator receives an image as input, which can be either real or generated by a generator. It processes this image using its layers of neurons, performing various operations such as convolution, activation, and pooling. At the last layer, the discriminator outputs the probability that the input image is real. If the probability is high, the discriminator considers the image to be real, if it is low, it is considered generated.

A discriminator's loss function measures how well it distinguishes between real and generated images. The smaller the value of the loss function, the better the discriminator copes with its task. The discriminator is trained by backpropagating the error, adjusting the neuron weights to minimize the loss function. This process continues until the discriminator learns to accurately distinguish between real and generated images.

Choosing the optimal architecture and tuning the discriminator parameters is critical for successfully training a GAN and generating high-quality camouflage patterns. The discriminator plays a key role in the GAN, acting as a critic and expert to help the generator improve its skills in creating realistic images.

IV. Problem Statement

The purpose of this work is to develop a method of generating military camouflage using GAN, which will allow creating a universal camouflage pattern that will effectively mask an object on a certain landscape from different angles.

To achieve this goal, an approach based on GAN training on a set of 10 photos of the same landscape taken from different angles will be used. This will allow the GAN to take into account the variety of visual characteristics of the landscape and create a camouflage pattern that will be effective when viewed from different angles.

As a generative-competitive network, it is proposed to use Conditional GAN (CGAN) (Fig. 3), which is an extension of the original GAN architecture, which allows you to control the image generation process with the help of additional conditional data. This conditional data can be in the form of class labels, text descriptions, images, or any other additional input provided to both the generator and the discriminator.

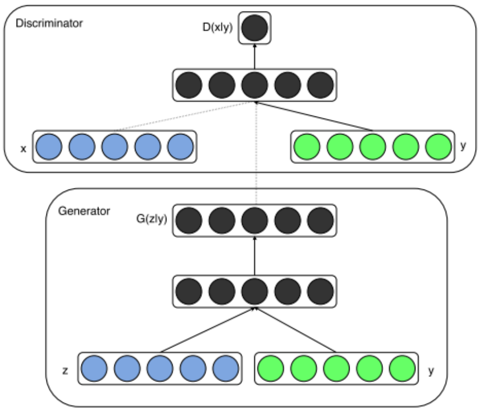

Fig. 3. Architecture Conditional GAN

The basic idea of CGAN is to guide the image generation process according to given conditional data. The generator attempts to generate images that match the given conditional data, and the discriminator must not only determine whether the generated image is realistic, but also verify that it matches the given conditional data.

A CGAN architecture typically consists of the following components.

- *Generator:* the CGAN generator takes as input a noise vector and conditional data (eg class labels or textual descriptions). The conditional data can be combined with the noise vector at different stages of the architecture, for example, at the input or in intermediate layers.
- *Discriminator:* the CGAN discriminator takes as input an image (real or generated) and conditional data. It must determine whether the image is realistic and whether it corresponds to the conditional data provided.
- *Loss function:* the CGAN loss function takes into account both the realism of the generated images and their correspondence to the conditional data. It can be an extension of the loss function of the original GAN or WGAN with an additional term that takes conditional data into account:

$$\min_G \max_D V(D,G) = \mathrm{E}_{x\sim p_{\text{data}}(x)}\left[\log D(x|y)\right] + \mathrm{E}_{z\sim p_z(z)}\left[\log\left(1-D\left(G(z|y)\right)\right)\right].$$

Conditional GAN is widely used in various applications where it is necessary to generate images with certain specified characteristics or attributes, such as:

1) Generation of images of objects of certain categories or classes.

2) Generation of face images with given attributes (gender, age, emotions, etc.).

3) Generation of images of landscapes or interiors with certain styles or lighting conditions.

4) Conversion of text descriptions into images (text-to-image synthesis).

However, successfully training CGANs can be challenging and requires careful tuning of the architecture and hyperparameters to ensure that the generated images match the conditional data. In addition, the quality of the results depends on the quality and completeness of the provided conditional data.

Pix2Pix, introduced by UC Berkeley researchers in 2016, is one of the most popular CGAN architectures (Fig. 4). It specializes in transforming images from one domain to another using paired data. This means that training Pix2Pix requires image pairs where one image is the input image and the other is the desired output image. This means that Pix2Pix can learn to transform, for example, black and white photos into color, segmentation masks into photorealistic images, or even maps into satellite imagery.

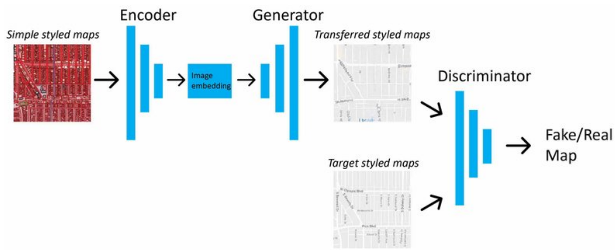

Fig. 4. Architecture Pix2Pix

Pix2Pix consists of two main components: a generator and a discriminator. The generator takes as input an image from one domain (for example, a black and white image) and tries to generate a corresponding image from another domain (for example, a color image). The discriminator evaluates the realism of the generated image, trying to distinguish it from a real image from the target domain.

During training, Pix2Pix minimizes two losses: adversarial loss, which forces the generator to produce images that the discriminator cannot distinguish from real ones, and L1 loss, which measures the difference between the generated image and its corresponding real image. This helps preserve the structure and content of the image during conversion.

Pix2Pix typically uses a U-Net-like architecture for the generator and a patch discriminator (PatchGAN) for the discriminator. The U-Net architecture is well-suited for image transformation tasks because it allows efficient transfer of information from low-level layers to high-level ones. PatchGAN evaluates image realism at the patch level, allowing it to focus on local details.

Pix2Pix is able to generate high-quality images with high resolution and realistic details, making it an ideal tool for creating camouflage that is difficult to distinguish from the real environment. By using L1 loss, Pix2Pix preserves the structure and content of the image during conversion. This allows you to create camouflage that not only masks the object, but also preserves its shape and contours. Pix2Pix can be used for many tasks, not only for creating camouflage. It can be used to colorize black and white photos, convert satellite images into maps, create photorealistic images from segmentation masks, etc.

However, Pix2Pix requires paired data for training, i.e. for each input image there must be a corresponding output image. This can be a limitation in some cases, especially when it is difficult to obtain sufficient paired data. Additionally, Pix2Pix may have difficulty performing complex transformations that require significant changes in image structure. For example, converting an image of a summer forest to a winter forest can be a difficult task for Pix2Pix.

V. Self-Attention GAN

Self-attention GAN (SAGAN) is an improved GAN architecture that uses a self-attention mechanism to better capture global dependencies in the image. The idea of self-attention is to allow the model to focus on different parts of the input image and take into account their relationships regardless of their distance (Fig. 5).

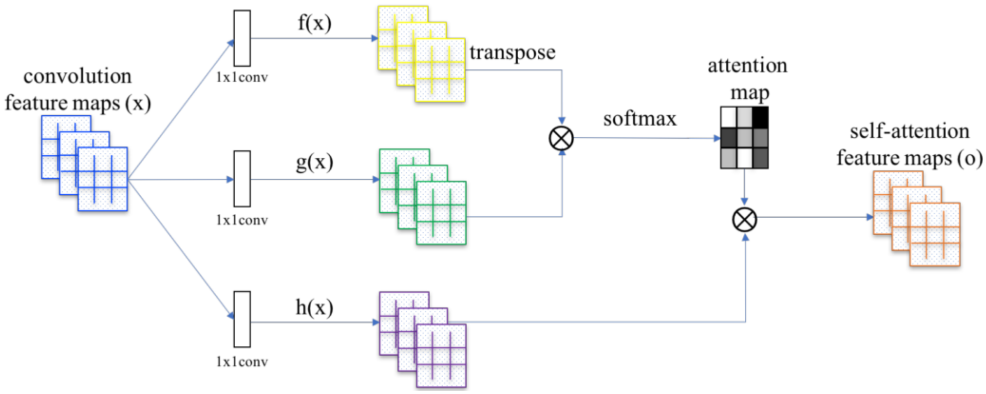

Fig. 5. Simplified architecture SAGAN

Self-attention GAN integrates a self-attention module into both the generator and the discriminator, allowing the model to efficiently capture long-term dependencies between pixels in an image. The key components of SAGAN are:

- *self-attention module*: This module calculates a weighted sum of the pixel values in the input image, where the weights are determined based on the similarity between pixels. This allows the model to focus on relevant parts of the image and take into account their relationships.
- *multi-level framework*: SAGAN uses a multi-level self-attention framework, where the self-attention module is applied at different image scales, allowing both local and global dependencies to be captured.
- *spectral normalization*: To stabilize training and improve performance, SAGAN uses spectral weight normalization instead of packet normalization.
- *enhanced attention*: SAGAN uses an enhanced attention mechanism that allows the model to focus on the most relevant parts of the image during generation.

Self-attention GAN has demonstrated superiority over previous GANs architectures in high-quality image generation tasks, especially for complex scenes and objects. The ability to capture global dependencies allows SAGAN to generate more coherent and detailed images. However, SAGAN has higher computational complexity compared to traditional GANs due to additional self-attention operations.

Self-attention GAN is able to generate images with high resolution and realistic details, thanks to the consideration of global dependencies. This is especially important for creating camouflage that must closely mimic natural textures and patterns. The self-attention mechanism allows SAGAN to generate complex and diverse structures that would be difficult to generate using traditional GANs. In addition, SAGAN exhibits greater stability in learning compared to traditional GANs.

However, the use of the self-attention mechanism leads to an increase in the computational complexity of the model, especially for high-resolution images. In addition, SAGAN, like other GANs, requires a large amount of data for efficient training.

VI. RESULTS

Analysis of the GAN training results shows a gradual improvement in the quality of the generated camouflage patterns with each epoch. At the initial stages of training (100 epochs), the pattern has high repeatability and insufficient diversity. As the number of epochs increases (200–500), the pattern becomes more complex, elements resembling natural objects such as tree branches, leaves and shadows appear.

However, even at the late stages of training (600–1000 epochs), the pattern still does not reach the desired level of realism. This is due to the limitation and heterogeneity of the input data. Using only 10 photos, even with magnification, may not provide enough variety to train the generator to produce high-quality and diverse camouflage patterns.

To further improve the results, it is recommended to expand the training data set to include more images with different angles, lighting and weather conditions. It is especially important to use more uniform images that have similar visual characteristics, such as color scheme, texture and structure. This will allow the generator to better generalize the features of the landscape and create more realistic and effective camouflage patterns (Figs 6 and 7).

VII. CONCLUSION

An intelligent system for generating camouflage patterns based on artificial intelligence technologies has been developed, which includes a conditional generative-competitive Pix2Pix network. Analysis of the training results of the generative-competitive network shows a gradual improvement in the quality of the generated camouflage patterns with each epoch. At the initial stages of training (100 epochs), the pattern has high repeatability and insufficient diversity. As the number of epochs increases (200-500), the pattern becomes more complex, elements resembling natural objects such as tree branches, leaves and shadows appear.

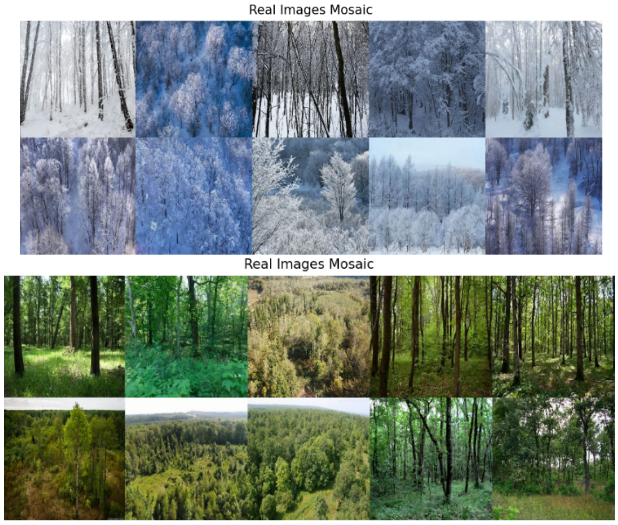

Fig. 6. Input images

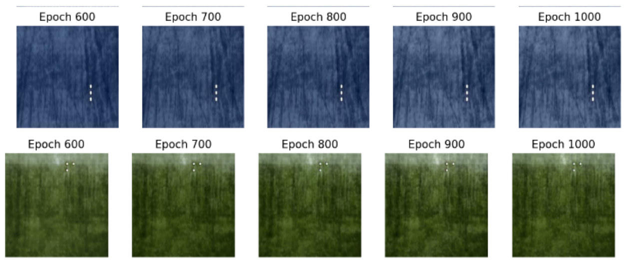

Fig. 7. Output images

REFERENCES

[1] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," *in Advances in neural information processing systems*, 2014.

[2] F. Chollet, *Deep learning with Python*. Manning Publications Co., 2017, pp. 364–380.

[3] J. Langr and V. Bok, *Generative adversarial networks (GANs) in action*. Manning Publications Co., 2019, pp. 10–35.

[4] A. Narita, K. Yoshioka, and D. J. Im, *Generative adversarial networks with industrial applications*. Springer, 2020.

Received March 07, 2024

Sineglazov Victor. ORCID 0000-0002-3297-9060. Doctor of Engineering Science. Professor. Head of the Department of Aviation Computer-Integrated Complexes.
Faculty of Air Navigation Electronics and Telecommunications, National Aviation University, Kyiv, Ukraine.
Education: Kyiv Polytechnic Institute, Kyiv, Ukraine, (1973).
Research area: Air Navigation, Air Traffic Control, Identification of Complex Systems, Wind/Solar power plant, artificial intelligence.
Publications: more than 700 papers.
E-mail: svm@nau.edu.ua

Nikulin Dmytro. Student.
Aviation Computer Integrated Complexes Department, Faculty of Air Navigation, Electronics and Telecommunications, National Aviation University, Kyiv, Ukraine.
Research interests: automatic control systems.
Publications: 1.
E-mail: dmytronikulin9@gmail.com

В. М. Синєглазов, Д. О. Нікулін. Інтелектуальна система генерації рисунків камуфляжу на основі технологій штучного інтелекту

Роботу присвячено розробленню інтелектуальної системи генерації рисунків камуфляжу на основі технологій штучного інтелекту. В якості інтелектуального елементу даної системи використовується генеративно-змагальна мережа. Для вирішення проблеми режиму колапсу використовується архітектура прогресивно зростаючих GAN (ProGAN). Система дозволяє генерувати абсолютно нові рисунки камуфляжу для обраної місцевості ітеративно покращуючи рисунок. За рахунок механізму обмежень можна зафіксувати бажані аспекти рисунку (кольорова гама, шаблон, кількість кольорів) з вже існуючого рисунка і пристосувати його до бажаної місцевості. Система передбачає можливість генерації мікропатернів на рисунках для покращення маскування на близьких дистанціях. Оцінюючи рисунок камуфляжу система враховує додаткові параметри, такі як ракурс (з землі та повітря), час та погода.

Ключові слова: штучні нейронні мережі; штучний інтелект; інтелектуальна система генерації; генеративно-змагальна мережа; прогресивно зростаючі GAN; камуфляжні рисунки.

Синєглазов Віктор Михайлович. ORCID 0000-0002-3297-9060. Доктор технічних наук. Професор. Завідувач кафедри авіаційних комп’ютерно-інтегрованих комплексів. Факультет аеронавігації, електроніки і телекомунікацій, Національний авіаційний університет, Київ, Україна.
Освіта: Київський політехнічний інститут, Київ, Україна, (1973).
Напрям наукової діяльності: аеронавігація, управління повітряним рухом, ідентифікація складних систем, вітроенергетичні установки, штучний інтелект.
Кількість публікацій: більше 670 наукових робіт.
E-mail: svm@nau.edu.ua

Нікулін Дмитро Олегович. Студент.
Кафедра авіаційних комп’ютерно-інтегрованих комплексів, Факультет аеронавігації, електроніки та телекомунікацій, Національний авіаційний університет, Київ, Україна.
Напрям наукової діяльності: системи автоматичного керування.
Кількість публікацій: 1.
E-mail: dmytronikulin9@gmail.com