Synthesis and Visualization of Image Datasets of Parametric 3D Model for Neural Network Training and Testing in Data-Poor Conditions

O.V. Kretinin[1,A], E.V. Popov[2,B], A.P. Tsapaev[3,A], L.O. Fedosova[4,A], M.I. Tyurikov[5,A]

[A] Nizhniy Novgorod State Technical University n.a. R. E. Alekseev
[B] Nizhny Novgorod State University of Architecture and Civil Engineering

[1] ORCID: 0000-0002-5672-9170, kretinin@list.ru
[2] ORCID: 0000-0002-3058-2369, popov_eugene@list.ru
[3] ORCID: 0000-0001-7336-4815, alexgrusp@mail.ru
[4] ORCID: 0000-0002-0635-3160, fedosovaludmila@list.ru
[5] ORCID: 0000-0003-1839-4506, sys32b@gmail.com

Abstract

Flaw detection of the inner surfaces of especially critical pipes is based on visualization, which requires complex optical systems equipped with artificial intelligence functions. Training of such systems is very complicated due to the limited volume of defective products. The paper describes training and testing of machine learning algorithms in poor data by the example of detecting a defect at the inner surface of a pipe. The authors propose a method for developing a set of synthetic training images obtained using 3D models of technical objects with parameterized defects applied to them. Images can be generated by parametric description of the artificially defected inner surface of a pipe 3D model in Autodesk Inventor environment. Windows AutoIt OS automation environment is applicable to generate synthetic images and masks. The method allows obtaining a set of synthetic pipe images for training neural networks with U-Net and LinkNet architectures. The trained neural networks testing has shown the defect recognition at a high level both on a synthetic sample of images and on real images of inner surface of rejected pipes.

Keywords: visualization, 3D modeling, machine learning, image processing, visual control, neural networks, data mining, U-Net, LinkNet.

1. Introduction

Computer vision and machine learning technologies have been widely applied to the monitoring of product facilities and quality assessment, as well as to flaw detection [11-14] in various fields, namely: agriculture [1-2], traffic management and road safety [3], construction and safety on construction sites [4-7], production planning [8], marketing [9], urban analytics [10] etc.

Machine learning of optical systems usually requires a sufficient number of visual images of the product surfaces together with defects. This approach to non-destructive testing of products by optical methods plays a significant role at the stage of product quality assessment and finds application in various production tasks. One of these tasks is to ensure the quality of pipes in the process of manufacturing products in the power industry.

When controlling the surface quality of technical objects in industry by automatic systems, there are two main problems based on the neural networks concept:

- insufficiency of real images with defects (including labeled ones) necessary for training of neural networks;

- the need to develop a well-founded procedure to assess the quality of computer vision systems, including for regulatory agencies.

One of the approaches to these problems solution is 3D model visualization of technical objects with parametrized artificial defects applied to them. These models allow creation of large sets of labeled surface images required for training neural networks. In addition, clear interrelation between the size and characteristics of defects with their images provides an evidence base confirming the performance quality of the computer vision systems.

This paper describes the method developed for adjusting neural networks to recognize and classify images of inner tube surface with artificial defects. The images dataset was created on the basis of 3D surface models with superimposed parameterized volumetric defects in the absence of real images of the object.

Studies were carried out on small-sized pipe models that are part of heat exchange units. Artificial defects were superimposed to the inner pipe surfaces.

2. Problem state

Feng Liu, Ronghang Zhu, Dan Zeng et al (2018) dedicate their paper [15] to the facial recognition method based on 2D images and their 3-dimensional reconstructions.

The paper by Xingchao Peng and Baochen Sun et al (2015) [16] describes an approach to the training of deep convolutional neural networks (DCNN) in the recognition of different objects using generated images based on 3D models. The article says that this approach can be effective when the set of real images is limited. Cars, airplanes, animals are used as objects for recognition. Different textures are applied to objects and backgrounds during the study process.

The article by Hassan Abu Alhaija, Siva Karthik Mustikovela, Lars Mescheder, Andreas Geiger (2018) [17] states that the success of deep learning in the field of computer vision is based on the availability of a significant volume of sets of labeled images. To reduce the need for manual image labeling, the use of virtual 3D objects is gaining popularity. The paper uses an approach based on the combination of synthetic and real images to recognize urban scenes observed when driving a car.

In general, we can note the high interest of machine learning algorithm developers in the use of synthetic images, including objects generated from 3D models to create training sets [20].

3. Training sampling

Computer algorithm that implements this method is shown in Figure 1 in the form of diagram. Roughly, the process of forming image datasets can be divided into the following stages:

Stage 1-creation of a 3D pipe model with defects in Autodesk Inventor;

Stage 2 - automatic generation of image dataset for a neural network based on a 3D pipe model by AutoIt;

Stage 3 - data preparation module for neural network;

Stage 4 - training of neural network based on generated synthetic image dataset;

Stage 5 - testing and evaluation of the neural network training.

At the 1st stage, one should create the pipe 3D model with given parameters (length, outer diameter, wall thickness) by for example Autodesk Inventor Professional software, which has a number of properties, namely:

- Autodesk Inventor is a parametric 3D modeling system;

- Autodesk Inventor has efficient 3D rendering module, which allows obtaining a realistic 2D screenshot based on 3D representation;

- the system includes such tools as macros and iLogic to automate the generating a large number of images of defects with given and programmable forms and dimension values.
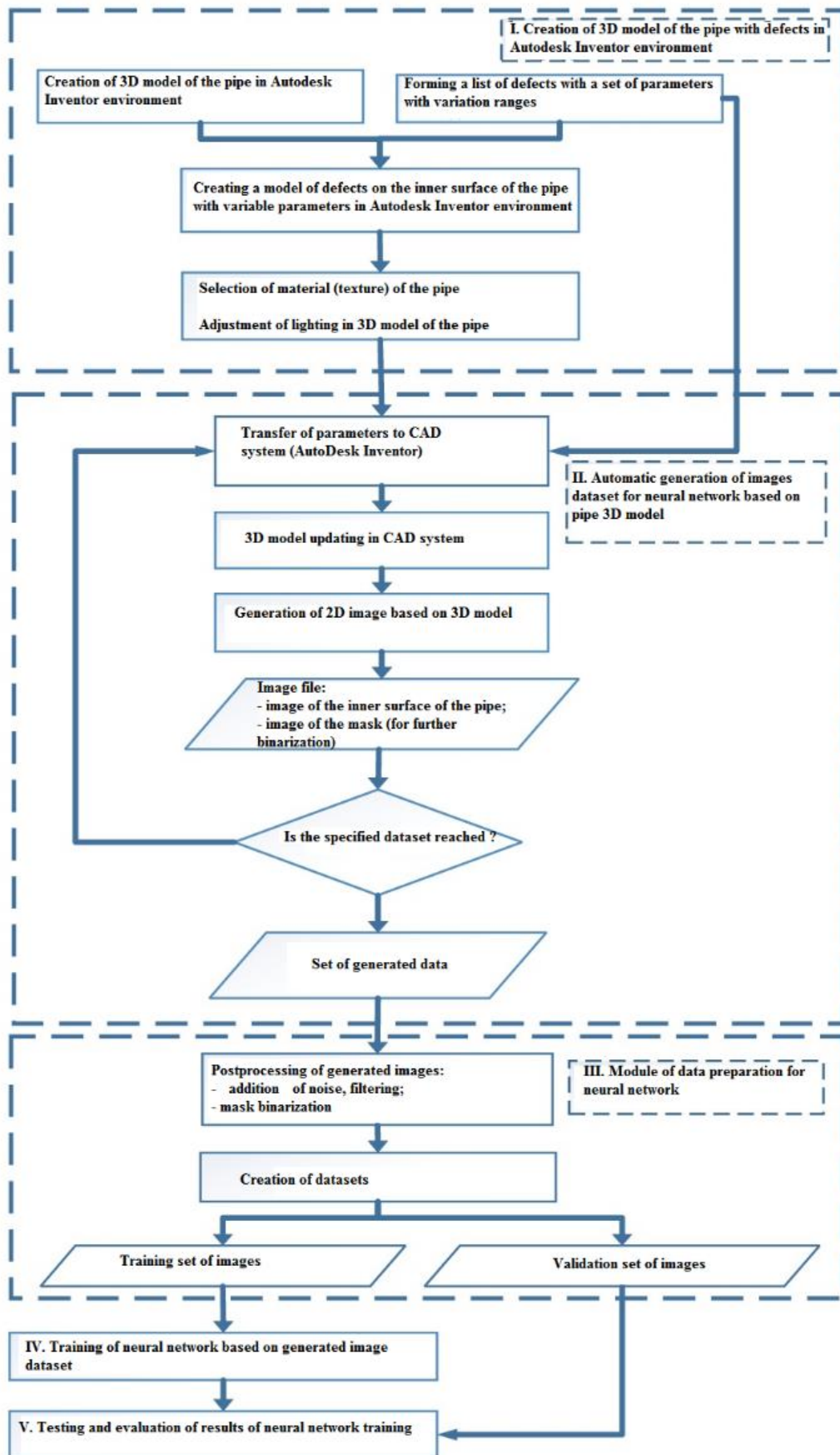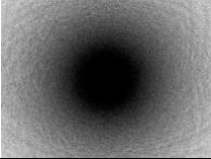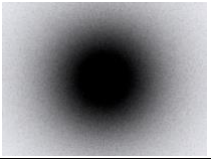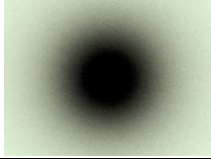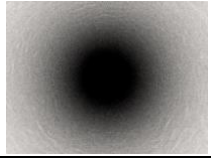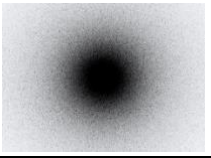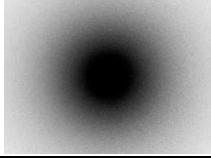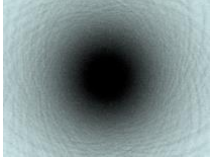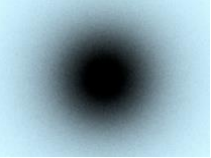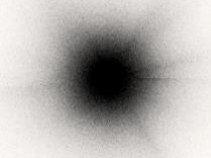
Fig. 1. Pipe 2D imaging algorithm based on 3D pipe model.

Further, it is very important to choose the appropriate textures for all objects in the scene to obtain a realistic view of the pipe inner surface. There is a base of standard pre-installed materials in Autodesk Inventor for this. Table 1 shows several options for choosing pipe material. To adjust the lighting, a directional light source is created with parameters (intensity 0-100%; attenuation compensation 1-100%; positional representation X mm, Y mm, Z mm).

Table 1. Visualization of pipe inner surface using material database.
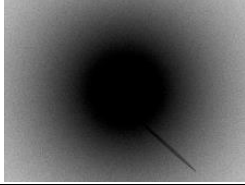
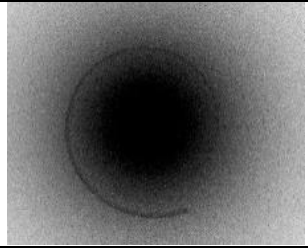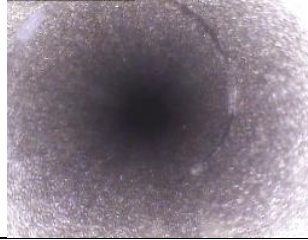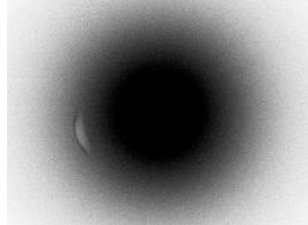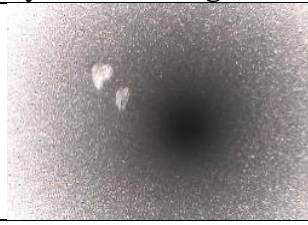|		
Polished steel	Silicon nitride (polished)	Titanium
Galvanized steel	Aluminium 1	Aluminium 2
Honed steel	Lead	Stainless steel (ground)
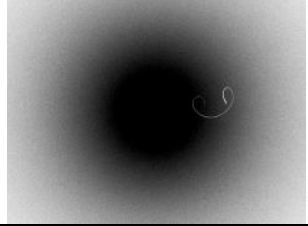
In order to form a set of synthetic defect images using the method of creating realistic 3D models, the most common defects (including foreign impurities detected on pipes) were selected: longitudinal guide mark, circular guide mark, handling mark, rolled blister and spatter. Using Autodesk Inventor there were created 3D defect models with variable parameters (see Figure 2 and Table 2).
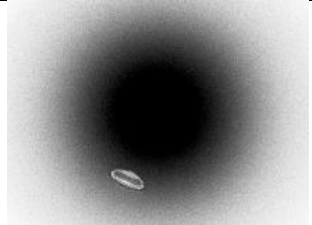
To create synthetic images, the mentioned group of defects can be expanded. Any kind of customer supplied defect can be selected. At the same time, the algorithm can be tuned to a defect other than the reference one. Thus, when using this system, a database of unknown defects will be accumulated, for which training samples can subsequently be created.

Stage 1 ends with the following results: 3D models of defects with parameterized variables and a list of defects and their parameters with ranges of their variation.

Table 2. List of defects with variable parameters

Longitudinal guide mark	
- displacement (determines the position of the defect on the inner surface of the pipe relative to its end face); - length of longitudinal guide mark; - cross-sectional diameter; - turn (determines the position of the defect on the surface of the pipe relative to its axis); - depth (defined as the distance from the axis of the pipe to its inner surface);	Synthetic Image
	Real Image

Circular guide mark	
- displacement (determines the position of the defect inside the pipe relative to its end face); - pitch of a helix; - cross-sectional diameter; - turn (determines the position of the defect on the surface of the pipe relative to its axis); - revolution (determines the length of the circular guide mark); - taper (determines the taper of the helix, forming the geometry of the circular guide mark); - helix radius (defines the radius of the circular guide mark);	 Synthetic Image Real Image
Handling mark	
- displacement (determines the position of the defect inside the pipe relative to its end face); - turn (determines the position of the defect on the surface of the pipe relative to its axis); - rotation (determines the position of the handling mark relative to its axis); - minor diameter of the handling mark; - major diameter of the handling mark; - depth of the handling mark (defined as the distance from the axis of the pipe to its inner surface); - interface (determines smoothness of defect boundaries);	 Synthetic Image Real Image
Rolled blister	
- displacement (determines the position of the defect inside the pipe relative to its end face); - turn (determines the position of the defect relative to the axis of the pipe); - cross-sectional diameter (defines the thickness of the rolled blister); - pitch (defines the pitch of the helix); - taper (determines the taper of the helix); - revolution (defines the length of the rolled blister); - depth of the dent (defined as the distance from the inner surface of the pipe to the axis of the rolled blister); - helix radius;	 Synthetic Image Real Image
Spatter	

- displacement (determines the position of the defect inside the pipe relative to its end face); - turn (determines the position of the defect on the surface of the pipe relative to its axis); - rotation (determines the position of the defect, relative to its axis); - diameter (shape-generating parameter)	 Synthetic Image Real Image

At the second stage, the problem of automatic generation of 2D images based on 3D models is solved. The AutoIt programming language is to be used as a means to automate the generation process. This language allows creating automation scripts (macros) that can simulate user actions, such as text input and operations with system and program controls, as well as respond to events. With the help of AutoIt, the following tasks are solved: control of the image generation automation system, organizing of operation and statistical information, as well as calculation required when creating new combinations of defect parameters. The algorithm for automatic generation of synthetic images is given in Figure 2.

To simplify the task of automating image generation, use the iLogic technology built into Autodesk Inventor. iLogic allows designing in Autodesk Inventor based on certain rules, so that user can automate and customize 3D models. Rules can be installed with the assembly or with an external file. With the use of iLogic rules and forms, import and export of parameters of a 3D model of the pipe with defects is performed.

The result of the second stage is the formation of data sets: synthetic images of the pipe with defects and binary masks.

At the third stage, post-processing of the generated synthetic images takes place, namely, the masks are binarized and additional noise is added to the image of the pipe with the defect. Next, a training set of images is formed, which is to be used for adjusting the neural network, and a validation set is formed, which is to be used for selection of the best settings of the neural network obtained during training.

4. Practical implementation

The neural network training script is implemented in Python using Keras and TensorFlow libraries to describe architecture and training and by OpenCV and Numpy libraries to load and preprocess images and masks from the training set. Training was carried out for 100 training epochs, with the preservation of the best weights, the Intersection over union (IoU) metric was used to evaluate the accuracy of neural networks, Adam was chosen as the optimization algorithm, the initial learning rate coefficient was chosen equal to 0.0001. The learning was divided into 2 stages: decoder training, so as not to damage the pre-trained encoder model with significant errors at the beginning of training, and training of the entire neural network.
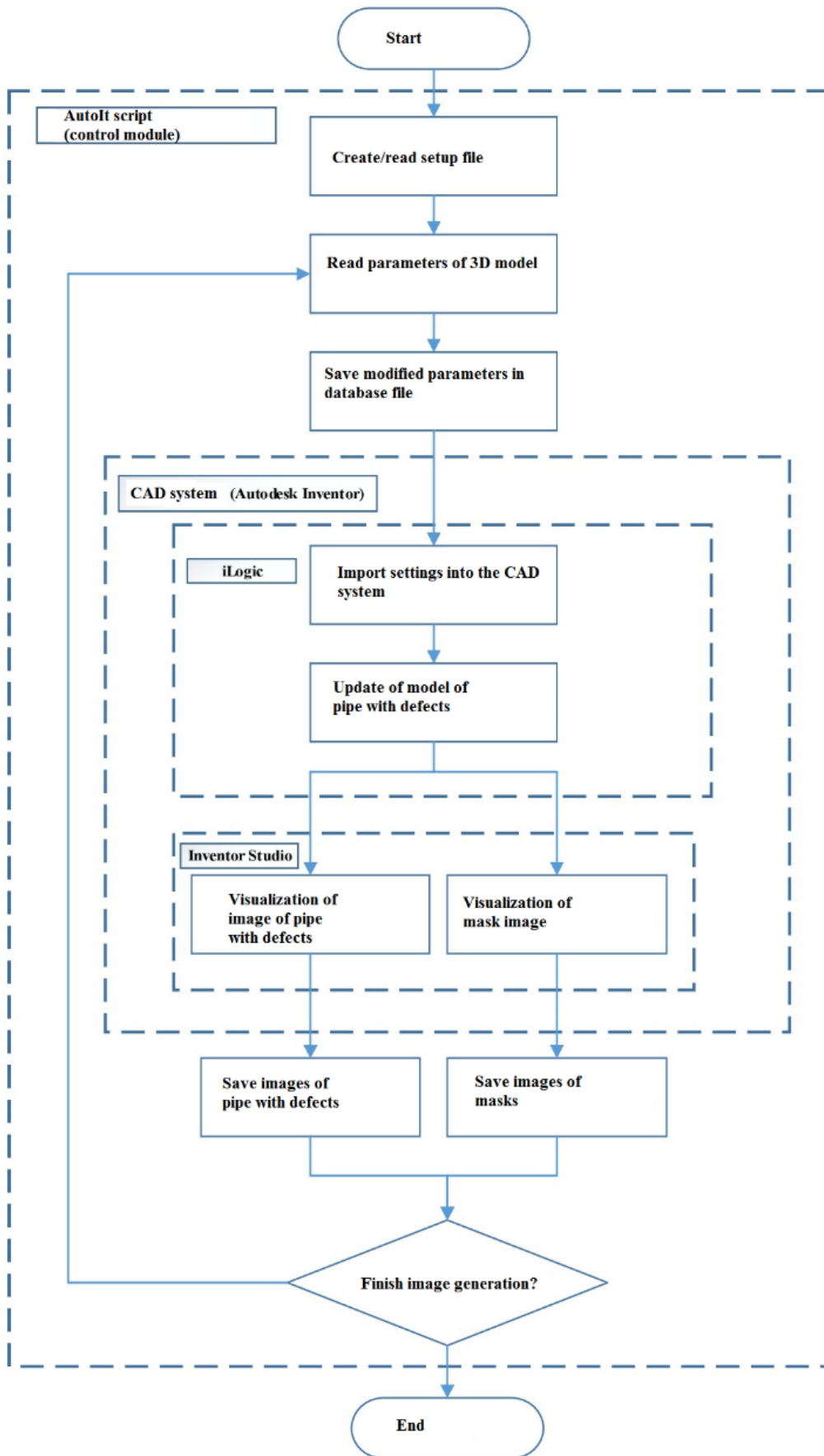
Fig 2. Algorithm for automatic generation of synthetic images based on 3D pipe model

To solve the problem of detecting defects, U-Net neural network, widely used in tackling segmentation issues [21.22] and a smaller scale LinkNet [23-24] neural network were chosen.

U-Net architecture is similar in its structure to the VGG classification neural network. To focus on areas with target objects, apart from contraction layers, U-Net also contains expansion blocks. The part of the neural network that is used to capture the context of the image while gradually reducing the image size is called encoder and essentially represents a neural network used for classification, but without layers that predict object classes in the image. The part of the neural network that is designed to generate masks and enables precise localization of the detected features is called a decoder.

A feature of the U-Net architecture (Fig. 3) is that encoder layers are connected to the equivalent in size decoder layers, due to which the boundaries of objects on the resulting masks are more precisely mapped, and this allows performing segmentation of objects of small sizes in images as well. Such connections are called skip connections and are used to pass features from the encoder path to the decoder path; in addition, such connections allow error gradients to approach the earlier layers of the neuron network without vanishing, which accelerates the neural network training process. The results of the U-Net neural network operation on test sets of synthetic images generated from 3-dimensional models of the pipe surface are given in Table 3.

Table 3. Results of u-net neural network operation

Neural network - U-Net (8,047,441 - number of training parameters, IoU coefficient = 0.8)				
Type of defect	Image of pipe with defects (1000 pcs.)		Image of pipe without defects (1000 pcs.)	
	NN detected defects	NN didn't detect defects	NN detected defects	NN didn't detect defects
Circular guide marks	77,5%	22,5%	0,2%	99,8%
Longitudinal guide marks	65%	35%	0,15%	99,85%
Handling marks	100%	0%	0,5%	99,5%
Rolled blister	92,5%	7,5%	1%	99%
Spatter	100%	0%	0,8%	99,2%

LinkNet is a faster neural network in comparison with U-Net. This is obtained by transforming the decoder part. In LinkNet, the combination of encoder and decoder features is accomplished by addition, as opposed to concatenation in U-Net, which results in fewer parameters and required calculations in subsequent layers (Figure 3).

In the case of both neural networks, the classification neural network MobileNet was chosen as the encoder for detecting defects. The choice is justified by a low number of parameters and, accordingly, low requirements for computing resources and fast learning ability.

To speed up training process, a MobileNet model, pre- trained on the ImageNet dataset was used. The results of the LinkNet neural network operation on test set of synthetic images generated from 3-dimensional models of the pipe surface are given in Table 4.
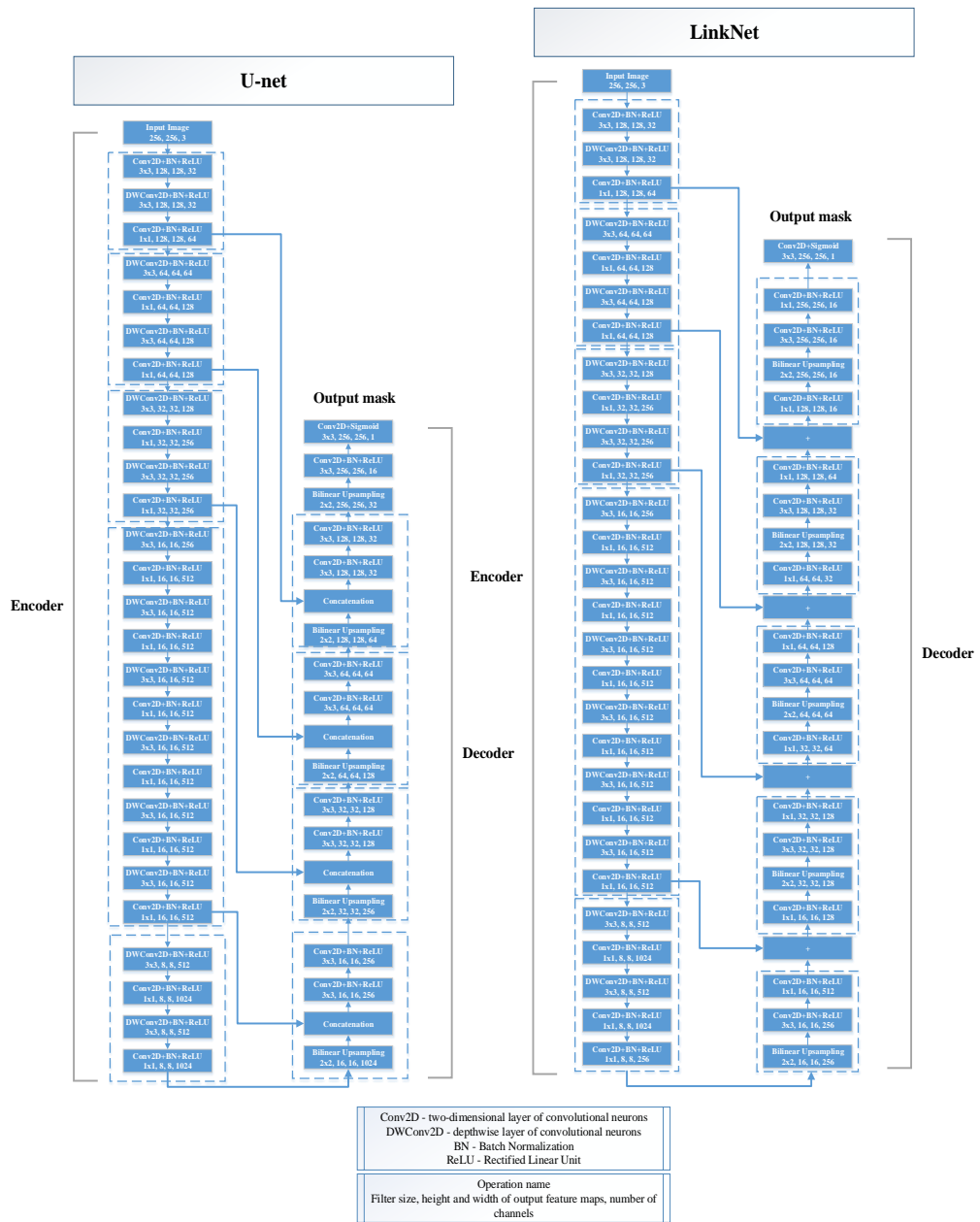
Fig. 3. U-net and LinkNet neural network architectures

Table 4. Results of the linknet neural network operation

Neural network - LinkNet (4 144 577 - number of training parameters, IoU coefficient = 0.8)				
Type of defect	Image of pipe with defects (1000 pcs.)		Image of pipe without defects (1000 pcs.)	
	NN detected defects	NN didn't detect defects	NN detected defects	NN didn't detect defects
Circular guide marks	67%	33%	0%	100%
Longitudinal guide marks	56,5%	43,5%	0%	100%
Handling marks	99%	1%	0%	100%
Rolled blister	91%	9%	1%	99%
Spatter	100%	0%	0%	100%

In Table 5, the examples of recognition of synthetic images of defects on the inner surface of a pipe generated from 3D models by U-Net and LinkNet neural networks are given.

The next step was testing of neural networks (U-Net and LinkNet) trained on synthetic images using real photographs of the internal surface of a pipe with defects (Table 6). In the process of detection of defects by neural networks, all defects found in real photographs were recognized, but the accuracy of recognition is to be enhanced, for example, by generating more realistic synthetic images for training.

Table 5. Examples of neural network defect detection

U-Net neural network		LinkNet neural network	
Synthetic image with defect	Result of recognition	Synthetic image with defect	Result of recognition

Table 6. Examples of defects detection by neural network based on real images.

Real image of inner surface of pipe with defect			
Result of recognition			
Type of defect	"longitudinal guide mark" defect type	"rolled blister" defect type	"handling mark" defect type
Real image of inner surface of pipe with defect			
Result of recognition			
Type of defect	"handling mark" defect type		"dust" defect type

5. Conclusion

The findings of the study:

1. The method of a training sample formation and visualization of a training set of synthetic images has been developed. Images are built on the basis of 3D models of technical objects with parameterized defects applied to them. This method is applicable in conditions of the lack of real images (including labeled ones) for training neural networks.

2. Pre-trained (following the proposed method) neural networks of U-NET and LinkNET architecture were tested in the task of detecting defects on synthetic images. Low rates of false detection of defects, less than 1%, as well as high rates of detection of defects such as "handling marks," "spatter," "guide marks": more than 91% were obtained.

3. U-Net and LinkNet neural networks, trained by the method proposed in the study, showed good level of recognizing defects on real images.

Moreover, it should be noted that this method allows simplifying the process of acquisition and labeling real data for the use in machine learning algorithms. It is especially relevant in tasks where the process of real data acquisition is difficult and there is lack of data for training.

Further development of the approach is aimed at expanding the base of defects and their synthetic models, reducing the time for generating synthetic images, developing a method for checking the adequacy of models, extending the method to other objects of industrial visual control.

References

1. Rafael Riederb, Computer vision and artificial intelligence in precision agriculture for grain crops: A systematic review, (Computers and Electronics in Agriculture, 2018), pp. 153:69-81.

2. Yuzhen Lu, Sierra Young, A survey of public datasets for computer vision tasks in precision agriculture, (Computers and Electronics in Agriculture, 2020), pp. 178.

3. Jae-Hong Kwon, Gi-Hyoug Cho, An examination of the intersection environment associated with perceived crash risk among school-aged children: using street-level imagery and computer vision, Accident, (Analysis and Prevention, 2016), pp. 97:111-121.

4. Hanbin Luo, Computer vision applications in construction safety assurance, (Automation in Construction, 2019), pp. 110:103013.

5. Weili Fang, Peter E.D. Love, Hanbin Luo, Lieyun Ding, Computer vision for behaviour-based safety in construction: A review and future directions, (Advanced Engineering Informatics, 2019), pp. 43.

6. Ipek Gursel Dino et al, Image-based construction of building energy models using computer vision, (Automation in Construction, 2020), pp. 116:103231.

7. Weili Fang and Ling Ma et al, Knowledge graph for identifying hazards on construction sites: Integrating computer vision with ontology, (Automation in Construction, 2020), pp. 119:103310.

8. Yuanbin Wang and Pai Zheng et al, Production planning for cloud-based additive manufacturing - A computer vision-based approach, (Robotics and Computer-Integrated Manufacturing, 2019), pp. 58:145-157.

9. Annemarie Nanne and Marjolijn Antheunis et al, The use of computer vision to analyze brand-related user generated image content, (Journal of Interactive Marketing, 2020), pp. 50.

10. Mohamed R Ibrahim, James Haworth, T. Cheng, Understanding cities with machine eyes: A review of deep computer vision in urban analytics, (Cities, 2019), pp. 96.

11. Luke Scime, Jack Beuth, Anomaly detection and classification in a laser powder bed additive manufacturing process using a trained computer vision algorithm, (Additive Manufacturing, 2017), pp. 19.

12. Zeqing Jin, Zhizhou Zhang, Grace Gu, Autonomous in-situ correction of fused deposition modeling printers using computer vision and deep learning, (Manufacturing Letters, 2019), pp. 22.

13. Dongming Feng, Maria Qing Feng, Computer vision for SHM of civil infrastructure: From dynamic response measurement to damage detection – A review, (Engineering Structures, 2018), pp. 156:105-117.

14. Shaun Falconer, Geir Grasmo, Ellen Nordgård-Hansen, Condition monitoring of HMPE fibre rope using computer vision during CBOS testing, (Conference: OIPEEC Conference, 2019).

15. Feng Liu and Ronghang Zhu et al, Disentangling Features in 3D Face Shapes for Joint Face Reconstruction and Recognition, (Conference on Computer Vision and Pattern Recognition, 2018).

16. Xingchao Peng and Baochen Sun et al, Learning Deep Object Detectors from 3D Models, (IEEE International Conference on Computer Vision (ICCV), 2015).

17. Hassan Abu Alhaija, Siva Karthik Mustikovela, Lars Mescheder, Andreas Geiger, Augmented Reality Meets Computer Vision : Efficient Data Generation for Urban Driving Scenes, (International Journal of Computer Vision, 2018).

18. D.V. Zinovyev, Principles of design in Autodesk Inventor 2016, 2nd ed., (edited by Azanov M.–M.: DMK Press, 2017).

19. Matheus C.Carvalho, Practical Laboratory Automation. Made Easy with AutoIt, (Publisher: Wiley-VCH/ ISBN: 978-3-527-34158-0, 2016).

20. J. Xie, M. Kiefel, M. T. Sun, A. Geiger, Semantic instance annotation of street scenes by 3D to 2D label transfer, (In Proceedings of IEEE conference on computer vision and pattern recognition (CVPR), 2016).

21. Bahareh Behboodi, Hassan Rivaz, Ultrasound segmentation using U-Net: learning from simulated data and testing on real data, (Conference: 41st Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC), 2019).

22. Yuncheng Zhou, Ke Zhang, Xinzhe Luo, Sihan Wang, Anatomy prior based U-net for pathology segmentation with attention, (School of Data Science (Fudan University), 2020).

23. Abhishek Chaurasia, Eugenio Culurciello, LinkNet: Exploiting encoder representations for efficient semantic segmentation, (School of Electrical and Computer Engineering (Purdue University), 1707.03718v1, 2017).

24. Lichen Zhou, Chuang Zhang, Ming Wu, D-LinkNet: LinkNet with Pretrained Encoder and Dilated Convolution for High Resolution Satellite Imagery Road Extraction, (IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2018).

25. An Tran, Ali Zonoozi, Jagannadan Varadarajan, Hannes Kruppa, PP-LinkNet: Improving Semantic Segmentation of High Resolution Satellite Imagery with Multi-stage Training, (SUMAC, arXiv:2010.06932v1, 2020).