

The study of the possibility of using innovative technologies for saving engineering 3d objects

Ekaterina Sokolova^{1,*}

¹NCIMM, Information Technologies Department, 362021 Vladikavkaz, Russia

Abstract. The prospect of using 3D printers in machine engineering is economically feasible since these devices significantly speed up the process of developing new products, substantially reduce the risks of design errors, reduce the cost of layout, and now their prices make them affordable for most world enterprises. The article studies the possibility of using innovative technologies for saving engineering 3D objects.

1 3D technologies in machine engineering

Car industry widely uses fast prototyping as an integral part of the design process. Obtaining prototypes by means of traditional methods (mechanical processing, casting) takes several weeks or even months and is a complicated, longtime and expensive design stage; contracts signing for the high-quality prototypes production is costly and may lead to a delay in the product release. The fast design cycle in the automated manufacturing process requires the use of up-to-date solutions based on fast prototyping, which allows to get items of almost any shape with different material properties quickly and inexpensively.

The prospect of using 3D printers is economically feasible since these devices significantly speed up the process of developing new products, substantially reduce the risks of design errors and the cost of layout, and now their prices make them affordable for most Russian enterprises. 3D printers with the best accuracy, quality, performance of prototyping, are successfully used in engineering. A 3D printer creates a prototype based on a 3D computer model created in any of the 3D CAD programs.

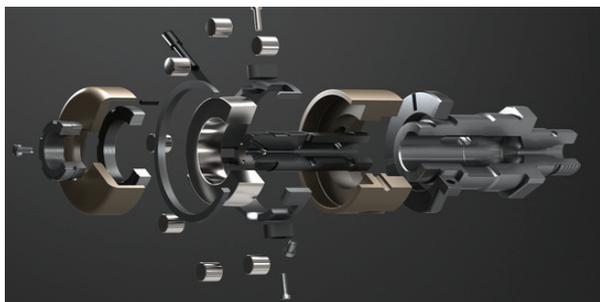


Fig. 1. 3D computer model

There are a lot of ways of prototyping usage, for example, you can print a transparent part of an engine

casing or a gearbox and check the oil circulation and the operation of the whole system.

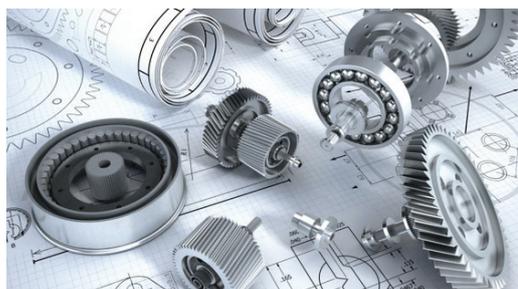


Fig. 2. 3D component parts

Also, there is the possibility of making models, which allows you to skip the stage of template production.

After scanning a cloud of points is obtained, which will need pre-processing, aligning, closing holes. At the output 3DS, ASE and STL file are obtained (other possible formats are dae, fbx, ma, txt, obj, ply, wrl, x3d, zpr).

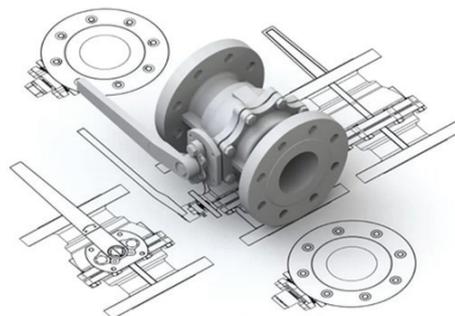


Fig. 3. An example of a 3D model of a machine engineering object.

Then the scanned part and the pre-prepared point cloud file are subject to the part's testing and verifying. This requires the 3D file of the model. Part verifying is

* Corresponding author: katya_sea@mail.ru

actually the process of creating a discrepancy card, i.e. a 3D colored diagram or several sectional views where one line shows the ideal part profile and the other shows the profile of the actual (scanned) part. Based on the above, the main problem in the process of saving three-dimensional models will be the algorithm that uses the selected format. In the case of using lossy algorithms, artefacts cannot be ruled out, which is unacceptable in the case of prototyping.

To solve the stated problem we propose the mathematical model that takes into account special features of 3D models.

2 Mathematical model development

It is required to develop a mathematical model for processing a 3D model, which will take into account the features of the 3D printing process.

To do that it is necessary to develop a compression method that will turn the pixels of the three-dimensional object into a bit set of points that will be shorter than the original digital representation.

Given the known information about the structure of images obtained as a result of 3D scanning, it was decided to use:

- system and information-and-communication approaches, various computer graphics methods, 3D modelling as well as the theory of sets. The program architecture was developed with the help of component-oriented methods and object-oriented programming.

The formation of a 3D model occurs by converting images from several cameras into a set of spatial points, which can be represented as a multidimensional array.

A specific feature here is the use of effective implementation of the known methods of group encoding to reduce the amount of the stored file without loss of quality, a new method of pixel transformation (bit set of values) that allows to foresee and prevent the appearance of artefacts when saving a file.

To solve the problem that arises due to the existence of the principle of constructing a multidimensional Cartesian space and to order points in a multidimensional space, the correspondence can be written in the following form:

$$P^n = P_1^1 \times P_1^1 \times \dots \times P_n^1 \quad (1)$$

where for every coordinate $P^n \in P^1$ there is a certain order.

Therefore, in order for all points to be ordered, it is necessary to introduce additional conditions and limitations (mathematical formulas, algorithms, etc.). Depending on the chosen limitation, the algorithm of points enumeration will be chosen; besides the larger the range, the less informative it will be.

The task of ordering the indicated points is carried out with the following condition:

- points adjacent to each other should not fall outside the range.

If you depict the entire object as consisting of lines, you can consider them as curves filling the space.

A formula can be determined for each of those curves:

$$I(k) = \sum_{j=1}^n j q_j(k) \quad (2)$$

For each k-th element of the sequence, which we take as $l(k)$, only one q function can be other than zero.

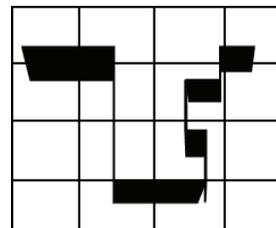


Fig. 4. An example of an object consisting of lines

Let's assume that x_0, y_0, z_0 is the relative displacement of the object if we take the global coordinate system as the basis, then:

$$x_0 = \min x_i, y_0 = \min y_i, z_0 = \min z_i$$

The lengths of the sides of the parallelepiped S_x, S_y, S_z are defined as follows:

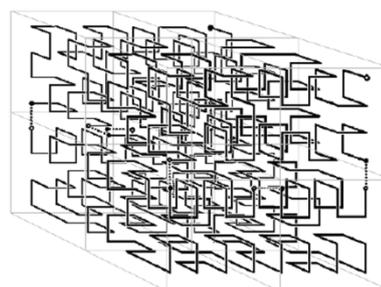


Fig. 5. Representation of an object in the form of lines in space

It is necessary to calculate the difference between the displacements and find the maximum value modulo.

Let's base on a three-dimensional coordinate system, which can be represented as a parallelepiped. Consequently, various transformations of the point cloud will be carried out in the selected coordinate system which includes the point cloud. The above procedure is adapted to the given task.

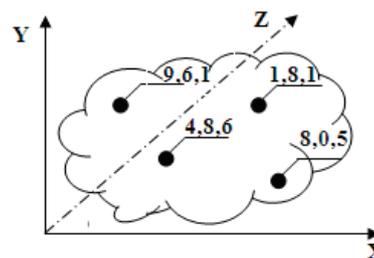


Fig. 6. Coordinates of a 3D object.

As you can see from figure 6, if you take any point in the point cloud, you can represent it in the Cartesian

coordinate system in case you choose relative positioning at the origin of the coordinates, taking into account the hardware specifics of the scanner.

Based on peculiarities of the practical implementation of the three-dimensional scanning process, it is possible to convert the initial set point cloud. With the newly introduced links between points, we can get a polygonal model. Let us denote the cloud of points as D , and the set of faces as V ; at that each face can be represented as a triangle with given vertices.

The model constructed above makes it possible to describe a set of points that do not have a specific structure, therefore there is excessive information on the object. Also, in case of changing the initial data, the model must be corrected using full enumeration of all the elements of the model.

Let us imagine the developed mathematical model, where D is the point cloud, G is the curve in which the order of points is described, and $R(G)$ is the function describing the principle of reference to points G . Below is its mathematical description:

Let us introduce the following definitions:

- O – 3D object;
- M – memory in bites;
- T – translation table;
- $T_b(i)$ – bit code position;
- B – block consisting of a three-bit union of positions;
- x, y, z – 3-bit code header in a block;
- T_D – displacement table;
- i, j, k – codes, additional bits, distance in the displacement table.

For each three-dimensional object, you need to specify a set of its vertices and their connections to each other.

Each pixel is represented by a bit code (Figure 3 shows a numerical interpretation of code values).

Then each of the codes must be placed in the byte memory while observing the condition: each following bit code is greater than the previous one.

The position of the bit code is constantly checked when stored in memory.

Then the obtained sequence is stored in the block, and at the same time the information is analyzed with the help of the construction of the bit tree and the displacement table based on it.

All of the above can be combined in the form of a mathematical model:

$$\left\{ \begin{array}{l} \forall O: \\ T \in M(b)T, \text{ where} \\ \forall i \in (1, m): Tb_{i-1} < Tb_i \\ B = b_x \cup b_y \cup b_z \\ \forall Ts(i, j, k) \equiv Tb \\ I(k) = \sum_{j=1}^n j q_j(k) \\ x_0 = \min x_i, y_0 = \min y_i, z_0 = \min z_i \end{array} \right. \quad (3)$$

To simplify the automation of the encoding process, we create the nodes of the binary tree by the example of a three-dimensional object based on 8 vertices $ABCD A_1 B_1 C_1 D_1$.

Each vertex has the following coordinates:

- A - 9,6,1
- B - 1,8,1
- C - 4,8,6
- D - 8,0,5
- A_1 - 0,7,7
- B_1 - 7,7,7
- C_1 - 4,4,0
- D_1 - 0,1,0

Let's make a gradation by the number of occurrences:

'1' -3, '4' -3, '5' -1, '6' -2, '7' -5, '8' -3, '9' -1, '0' -5

Then let's order them in ascending order:

9,5,6,8,4,1,7,0.

Let's start building a binary tree:



Fig. 7. Sorting vertices by number of mentioning.

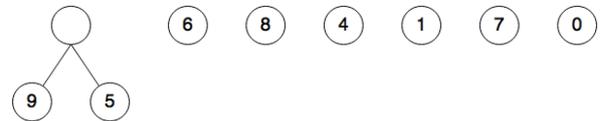


Fig. 8. Building the 1st layer of the tree.

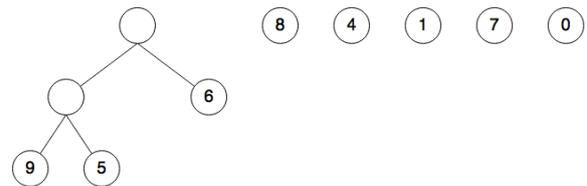


Fig. 9. Building the 2nd layer of the tree.

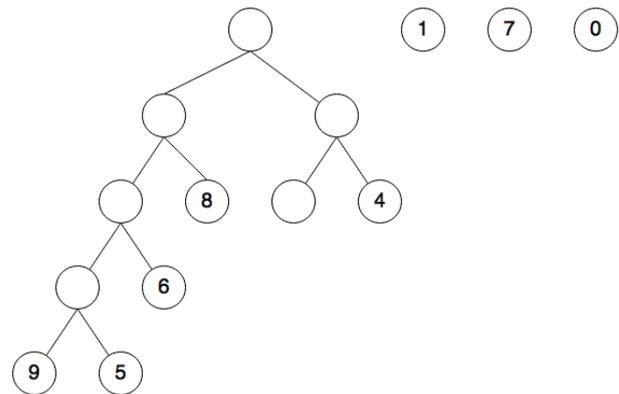


Fig. 10. Building the 3rd layer of the tree

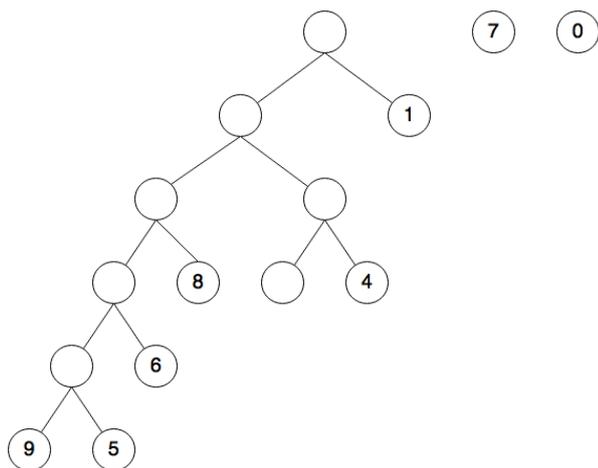


Fig. 11. Building the 4th layer of the tree.

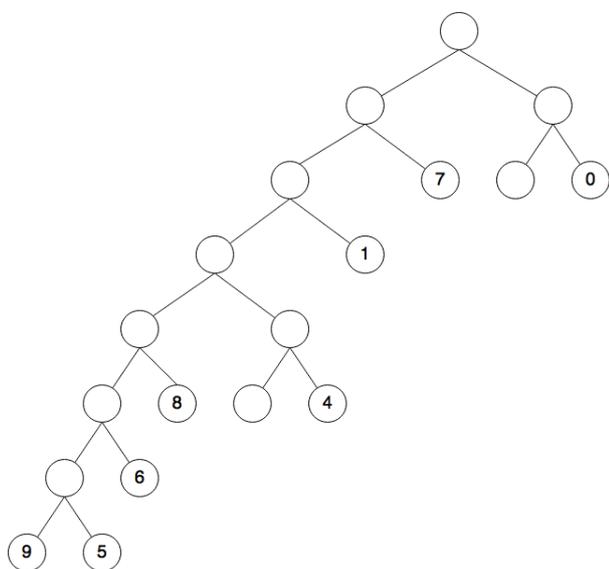


Fig. 12. Building the 5th layer of the tree.

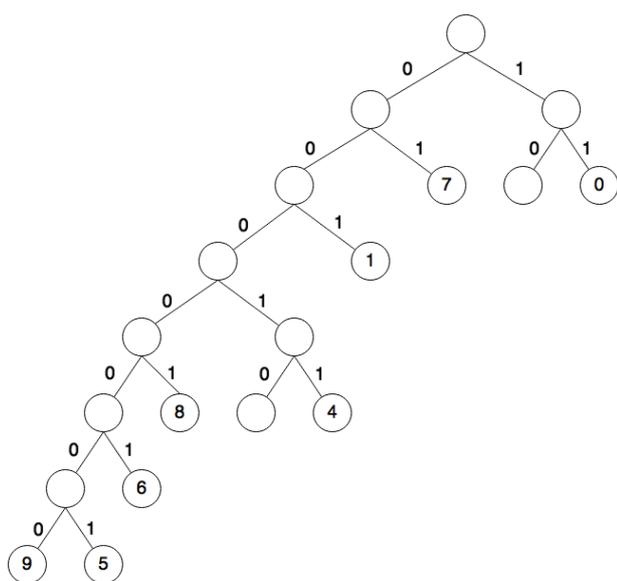


Fig. 13. Binary tree

3 Experiments

Quantitative characteristics of the product:
 - the mathematical model will allow to process 3D images (*.obj, *.3ds, *.ase);
 - the operating time of the mathematical model depends on the volume of processed images (directly proportional to the size of the downloaded file).

We conducted a number of experiments on the basis of which we constructed a diagram showing the dependence of the operation time of the mathematical model on the file size (Figure 14).

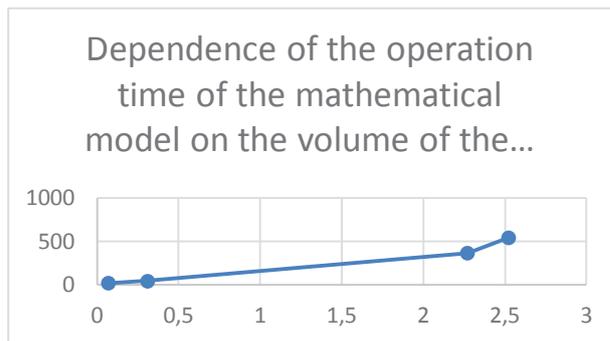


Fig. 14. Graph of the dependence of the operation time of the mathematical model on the volume of the processed images

The created lossless compression method for 3D models images for engineering will allow to obtain images without the appearance of artefacts with the probability of 0.87.

The appearance of artefacts directly depends on the number of restored points of the object. Figure 15 shows histograms representing the percentage of restored object points and the likelihood of artefacts.

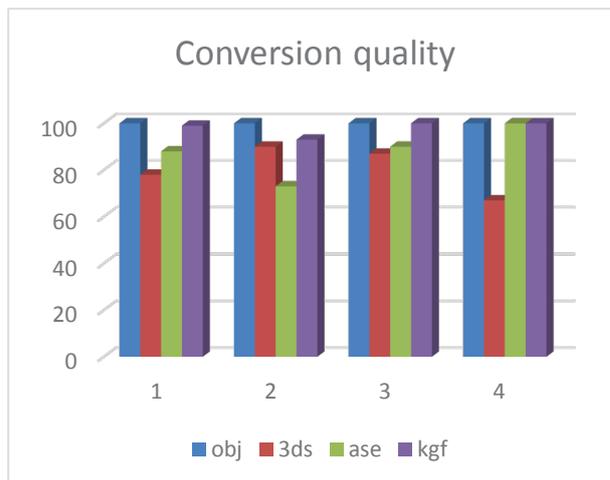


Fig. 15. Likelihood of artefacts

4 Conclusion

As a result of experimental studies, the possibility of using innovative technology for 3D engineering objects storing was confirmed. A fast design cycle in automated manufacturing process requires the use of up-to-date solutions based on fast prototyping, which allows to get

products of almost any shape with different material properties quickly and inexpensively. Taking into account the proposed mathematical model it also became possible to reduce the size of the stored models and avoid the appearance of artefacts.

References

1. Zhi-liang Zhu, Wei Zhang, Kwok-wo Wong and Hai Yu, *Inf. Sci.* **6** 1171-86 (2011)
2. Wei Zhang, Kwok-wo Wong, Hai Yu and Zhi-liang Zhu *Commun, Nonlinear Sci. Numer. Simul.* **3** 584-600 (2013)
3. Xingyuan Wang, Lin Teng and Xue Qin, *Signal Process* **4** 1101-08 (2012)
4. Yicong Zhou, Weijia Cao, and Philip Chen, *Signal Process* **8** 197-207 (2014)
5. A.A. Sokolov, A.S. Miroshnikov, E.A. Sokolova, *Gornyi Zhurnal* **12** 83-86 (2016)
6. M.A. Murillo-Escobar, C Cruz-Hernández, F Abundiz-Pérez, R.M. López-Gutiérrez and O.R. Acosta Del Campo, *Signal Process* **109** 119-131 (2015)
7. A.M. Kumaritov, E.A. Sokolova, A.A. Sokolov, *Gornyi Zhurnal* **2** 94-96 (2016)
8. Yingqian Zhang and Xingyuan Wang, *Inf. Sci.* **20** 329-351 (2014)
9. Wei Zhang, Kwok-wo Wong, Hai Yu and Zhi-liang Zhu *Commun, Nonlinear Sci. Numer. Simul.* **8** 2066-80 (2013)