DOI: 10.32703/2617-9040-2025-45-6

UDC 004.925.5

Oleksandr Gertsiy[1], Serhii Karnatov[2], Vitaliy Gladish[3], Valentyna Tkachenko[4]

[1]Assistant professor, Automation and Computer-Integrated Transport Technologies Department, National Transport University, M. Omelyanovicha-Pavlenko St., 1, Kyiv, 01010, Ukraine. ORCID: https://orcid.org/0000-0002-8634-5880.

[2]Postgraduate student, Automation and Computer-Integrated Transport Technologies Department, National Transport University, M. Omelyanovicha-Pavlenko St., 1, Kyiv, 01010, Ukraine. ORCID: https://orcid.org/0009-0006-6254-6166.

[3]Postgraduate student, Automation and Computer-Integrated Transport Technologies Department, National Transport University, M. Omelyanovicha-Pavlenko St., 1, Kyiv, 01010, Ukraine. ORCID: https://orcid.org/0009-0003-5801-056X.

[4]Assistant professor, Associate Professor, Department of Transport Law and Logistics, National Transport University, 2, Mykhailo Omelyanovich-Pavlenko str, Kyiv, 01010, Ukraine. ORCID: https://orcid.org/0000-0001-5147-0772.

Corresponding author: gertsiy_oa@gsuite.duit.edu.ua.

Modeling an Image Clustering Algorithm For Detecting Overheated Railway Axle

This paper presents an approach to image identification based on a clustering algorithm for detecting the thermal characteristics of railway axle boxes. The study analyzes the potential of clustering algorithms in the task of digital image identification. The principles of the proposed algorithm are described in the context of image segmentation and compression, as well as pattern recognition in the transportation domain. A block diagram of the algorithm is provided along with an explanation of its operational principles. A scheme for implementing a machine vision system using the proposed algorithm is suggested for the detection of overheated axle boxes in railway transport. Experimental modeling was conducted for image segmentation based on color models in the infrared spectrum to identify regions with elevated temperatures in the MATLAB environment. For the experiment, thermal images of rolling stock axle boxes were selected one representing a normal condition and the other representing an overheated condition. The simulation results demonstrated that segmentation based on image color models allows for accurate delineation of the thermal characteristics of axle box images. The study confirmed that the use of the proposed algorithm and its software implementation for thermographic cameras is effective for recognizing temperature indicators. Additionally, the algorithm enables image compression, which increases data processing speed and facilitates the monitoring of thermal characteristics of rolling stock at high speeds.

Keywords: *thermographic images, clustering, segmentation, monitoring, modeling, temperature indicators, axle box.*

Introduction. Currently, the field of graphical image identification has gained significant popularity among researchers and scientists. This is due to the growing number of practical applications of such tasks in everyday human activity. Examples include text recognition, fingerprint identification, license plate detection, facial recognition, and more. An important factor contributing to the popularity of image identification is the continuous development of theoretical and mathematical frameworks [1]. Among the key approaches for solving recognition problems are: classification using decision functions and distance metrics; support vector machines; neural networks; various statistical methods; and clustering algorithms.

In computer science, cluster analysis can be used to detect boundaries or recognize objects by dividing a digital image into distinct regions [2].

In the transportation sector, cluster analysis can be applied in pattern recognition for supporting control, operations, and modeling of transportation systems [3]. For instance, it can be used to determine scenarios in decision support systems based on traffic simulation, where scenario clustering procedures may assist in real-time weather-responsive traffic management (WRTM) by rapidly classifying current or forecasted weather conditions into predefined categories and suggesting appropriate WRTM strategies that can be tested through real-time traffic simulation prior to deployment [4]. Additionally, clustering algorithms can be used in image processing and visualization systems to assess railway traffic safety [5].

Analysis of recent research and problem statement. Clustering, or cluster analysis, in the context of identification tasks, refers to the process of partitioning a given sample of objects into non-overlapping subsets (clusters) such that each cluster consists of similar objects. The similarity between objects is determined based on a selected metric, which is chosen according to the clustering criterion.

The input data for the clustering procedure is a set of objects, each defined by a vector of features. In such algorithms, the vectors typically represent pixels or neighborhoods of pixels. The vector set may include the following components [1, 2]:

- intensity values;
- color codes or color characteristics;
- computed feature descriptors;
- textural characteristics.

The clustering algorithm itself can be described as a function a: X→Y, which maps any object $x \in X$ to a cluster number $y \in Y$. In some cases, the set Y (i.e., the number of clusters) is known in advance; however, more often, the objective is to determine the optimal number of clusters according to a specific clustering quality criterion.

Nevertheless, the solution to the clustering problem is inherently ambiguous due to several reasons:

- there is no universally optimal clustering quality criterion;
- the number of clusters is generally unknown and must be determined based on some subjective criteria;
- the result of clustering strongly depends on the chosen metric, which is also typically subjective and defined by an expert [6].

This ambiguity necessitates flexible data processing tools, which must be considered during the development of methods for graphical image identification.

Researchers and scientists worldwide are actively working on this issue within the domain of digital image processing. In particular, the segmentation of color images based on clustering for identification purposes has gained widespread popularity [7].

For example, in the railway industry, a well-known system for detecting overheated axle boxes is widely used for inspecting wagons and locomotives to detect technical malfunctions, especially overheated axle boxes [8]. The operational principle of this equipment is based on measuring the temperature of axle boxes as trains pass by infrared sensors. However, this system has several drawbacks: detecting overheated axle boxes may trigger false alarms when transporting hot cargo (e.g., sinter pellets), in the event of hot water leakage from passenger car heating systems, or when passing through the system while braking with friction brakes (due to heating of the brake pads) [9].

In one of the studies, the authors present a machine vision system for efficient monitoring, analysis, and representation of visual data obtained from multiple cameras [10]. This solution aims to enhance the safety of daily railway transportation in two key ways:

1. by evaluating a wide range of safety requirements using image analysis algorithms capable of processing large-scale train images, and
2. by assisting train safety operators in detecting any potential malfunctions within the train. The system utilizes high-speed visible and thermal cameras that monitor the train as it passes beneath a railway frame structure.

Therefore, instead of using outdated systems for detecting overheated axle boxes, technical vision systems based on thermographic cameras could be implemented [11]. Thermographic cameras detect radiation in the infrared range of the electromagnetic spectrum (approximately 0,9–14 µm) and generate images based on this radiation, which allows for the identification of overheated or undercooled objects. In such images, color serves as an indicator of temperature level. For example, blue indicates that the object's temperature is around 27-28 °C, violet indicates 29-31 °C, orange indicates 32-34 °C, yellow indicates 35-36 °C, and white represents temperatures above 36 °C, etc.

The advantages of such cameras include the following:

- the ability to display visual images, which aids in comparing temperatures across large surfaces;
- the capability to detect malfunctioning components before failure occurs;
- effective temperature measurement in areas where other methods are either impossible (e.g., low thermal capacity objects) or pose health risks;
- non-destructive testing;
- facilitation of defect detection (such as cracks) in columns or other metallic parts;
- the ability to visualize heat at a rate of one frame per second, even with relatively low spatial resolution [12].

To implement a machine vision system for detecting overheated axle boxes on railways, a structural diagram is presented in Fig. 1. The operating principle of this system is as follows: when a wheelset approaches sensor S1, it sends a signal to the control system CS (which records the serial number of the wheelset) and to the thermographic cameras C1 and C2 to initiate image capture. The cameras process the thermal images and, depending on the results, send a high-level signal to the CS if an excessive temperature is detected. The CS associates the wheelset number with the side of the axle box, logs the inspection results into the database, and for instance sends a signal to a controller, which in turn marks the axle box with paint or sends a signal to the train operator to reduce speed, and so on [13, 14].

Thus, the key challenge is the development of a high-quality image recognition algorithm - specifically for thermographic images that is both fast-acting and effective in the presence of other thermal interferences typically found in railway environments.

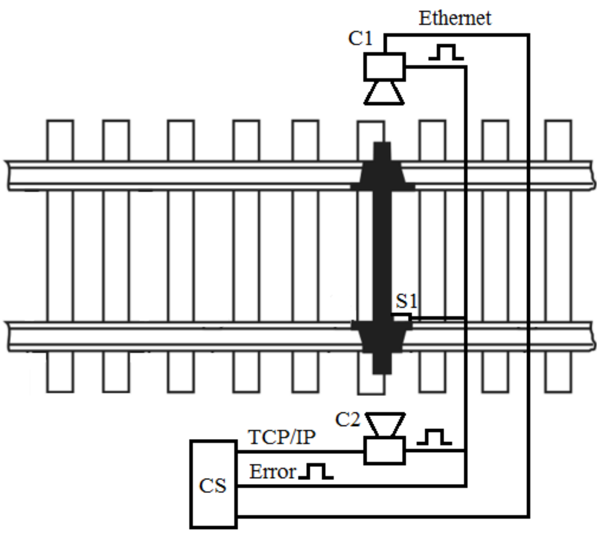


Fig. 1. **Structural Diagram of the Implementation of a Machine Vision System for Detecting Overheated Axle Boxes in Railway Transport**

The purpose and tasks of the study. To analyze the operating principle of the k-means clustering algorithm in image recognition tasks, and to conduct experimental modeling of this algorithm by segmenting thermal images based on color in order to recognize the thermal characteristics of the studied objects using the MATLAB environment.

Materials and methods of research. The k-means clustering algorithm partitions a set of elements in an n - dimensional vector space into k clusters, in which each object belongs to the cluster with the nearest mean value. This method yields exactly k distinct clusters with the greatest possible separation. The optimal number of clusters k, which results in the highest degree of separation, is not known a priori and must therefore be derived from the data. The goal of k-means clustering is to minimize the total intra-cluster variance or the sum of squared error (SSE) function [15, 16]:

$$J = \sum_{j=1}^{k} \sum_{i=1}^{n} \left\| x_i^{(j)} - c_j \right\|^2 , \tag{1}$$

where J – objective function;
k – number of clusters;
n – number of observations;
$x_i^{(j)}$ – observation i;
c_j – centroid of the cluster j;
$\left\| x_i^{(j)} - c_j \right\|^2$ – distance function.

Algorithm Description. The process begins by assigning objects to clusters. The number of clusters k is selected, and at the initial step, these points are considered the "centers" of the clusters. Each cluster corresponds to one center. The initial centroids can be selected in one of the following ways:

- selecting k observations to maximize the initial distance;
- randomly selecting k observations;
- selecting the first k observations.

As a result, each object is assigned to a specific cluster. Next, an iterative process begins. The cluster centers are recalculated and are considered to be the coordinate-wise means of their respective clusters. The objects are then reassigned based on the updated centers.

The simulation will be carried out in the Jupyter Notebook environment using the Scikit-learn library.

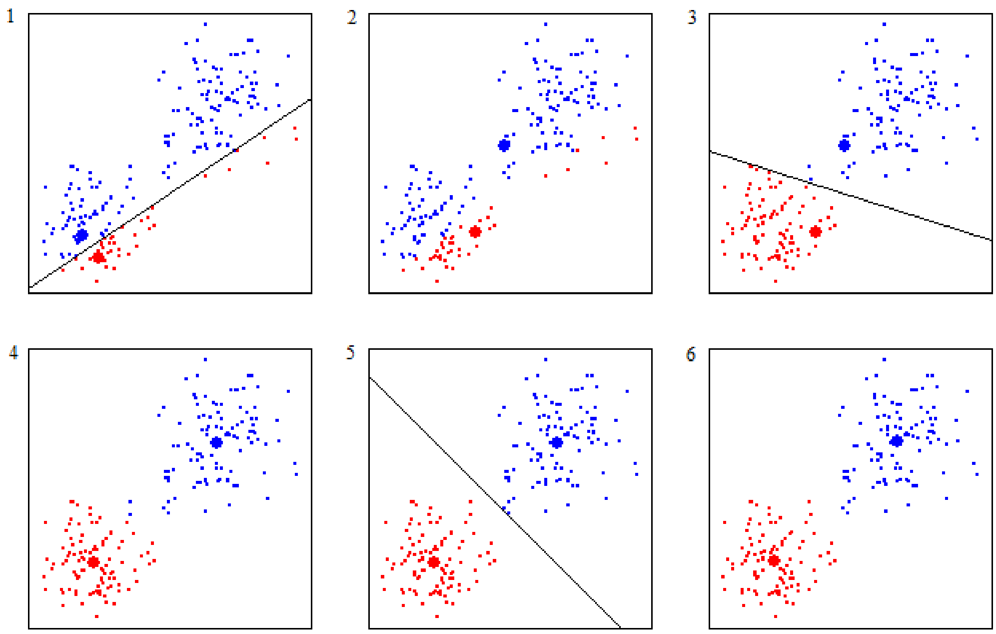


Fig. 2. **Illustration of the Operation of the K-Means Algorithm**

This process of computing cluster centers and redistributing objects continues until one of the following conditions is met:

- the cluster centers have stabilized, i.e., all observations belong to the same cluster as in the previous iteration;
- the number of iterations reaches the predefined maximum.

Figures 2 and 3 illustrate the working principle of the *k*-means algorithm (where *k*=2) and its block diagram [14, 17].

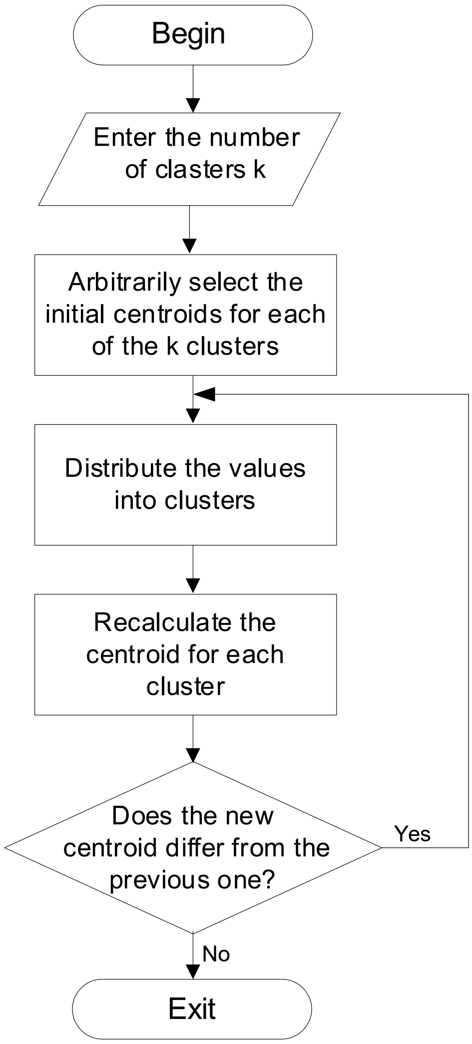


Fig. 3. **Block diagram of the k-means algorithm**

The simulation (Table 1) illustrates how the algorithm works on synthetic datasets. The essence of the simulation is as follows:

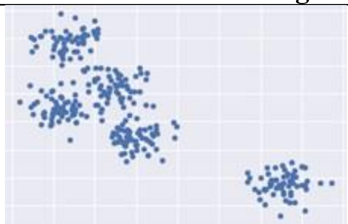
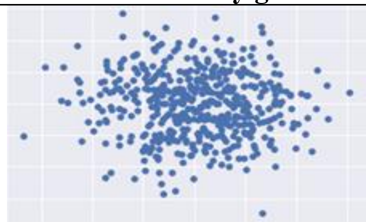
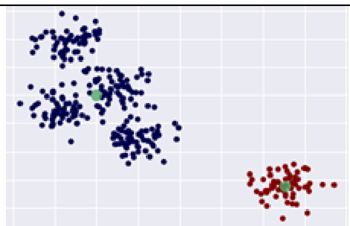
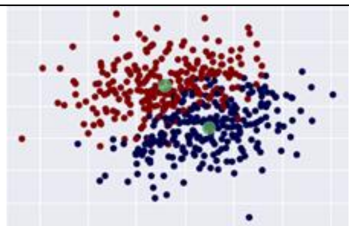
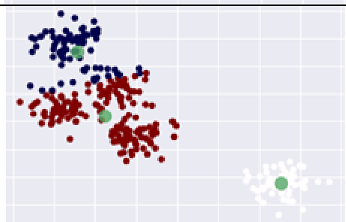
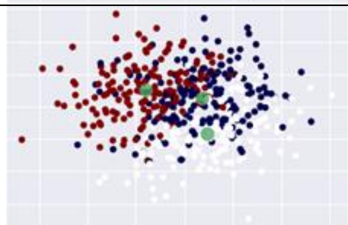
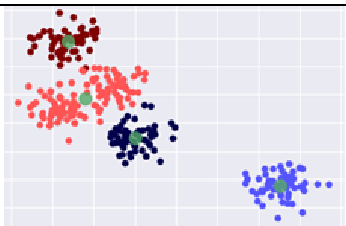
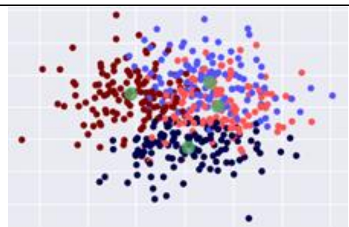
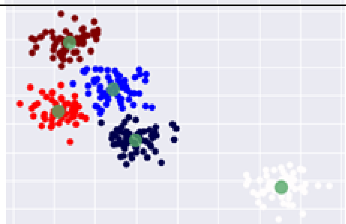
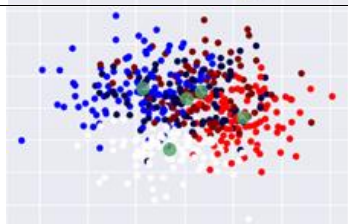
- in the Jupyter Notebook environment, four datasets are generated using the scikit-learn package, namely: datasets.make_blobs, datasets.make_gaussian_quantiles, datasets.make_circles, and datasets.make_moons;

- we apply the K-means algorithm, which uses the Euclidean distance metric, to each data type, and gradually increase the number of clusters from 2 to 5 to see how the algorithm behaves on different data types with different values of the k parameter.

Note. The number of clusters does not exceed 5, since we know in advance that the number of ‘correct’ clusters, for example, for data of type datasets.make_blobs is 5, in other cases no more than 2.

From Table 1, we can conclude that the K-means algorithm performed best for data of the datasets.make_blobs type with K=5. Each cluster is defined solely by its centre, which means that each cluster has a convex shape. As a result, the K-means algorithm can describe relatively simple shapes. In addition, the K-means algorithm assumes that all clusters have the same ‘diameter’ in a sense, it always draws the boundary between clusters so that it passes exactly in the middle between the cluster centres. This can sometimes lead to unexpected results, as shown in Table 1 on the datasets.make_gaussian_quantiles data.

Table 1. **Illustration of how the K-means algorithm works with artificially generated data**

Type of input data / Number of clusters	 datasets.make_blobs	 datasets.make_gaussian_quantiles
K = 2		
K = 3		
K = 4		
K = 5		

Segmentation of an object image consists in dividing it into spatially connected areas (segments). Global and local pixel connectivity maxima of quantized images are determined, and then regions adjacent to these maxima are formed. The set of these regions forms a spatial set of objects. In the process of image segmentation, several incorrectly classified elements (the so-called speckle noise) inevitably appear. After thresholding, such errors can be eliminated. If the brightness level of any image element is higher than the maximum level of its neighbors, then it is likely due to noise. It can be replaced by the maximum brightness level of neighboring elements. A similar operation, filtering, can be applied if the brightness level of an element is lower than the minimum level of its neighbors. Another possibility is to replace each level value with the average or median value of neighboring elements.

Let's consider the procedure for segmenting objects by determining the connectivity of image elements [18-21]. The mask matrix formed by selecting each minimal element will be called a cut, and the total connectivity value of a cut element will be called internal cut connectivity, and the total connectivity value between elements of neighboring cuts will be called intercut connectivity.

The value of internal cut connectivity $f(k)$ will be determined by the formula.

$$f(k)=\sum_{j=1}^{M-1}\sum_{i=1}^{N-1}\left(a_{ij}^{k}\wedge a_{ij+1}^{k}+a_{ij}^{k}\wedge a_{ij-1}^{k}+a_{ij}^{k}\wedge a_{i-1j}^{k}+\ldots+a_{ij}^{k}\wedge a_{i+1j+1}^{k}\right). \quad (2)$$

Next, we determine the global and local values of internal cut connectivity. The values of internal cut connectivity are determined at the time points t_{k_1-1}, t_{k_2}, t_{k_3+1}. These time points correspond to cuts $k_1-1,\ k_2,\ k_3+1$. The number of such cuts is $v=\overline{1,w}$, which corresponds to the general case. Let us define the functions of internal cut connectivity as follows:

$$f(k_v,k_{v+1})=f(k_v,k_{v+1}),\ldots,f(k_{v+p1-1},k_{v+p1}),\ldots,f(k_{vl_1-1},k_{vl_1}),\ p_1=\overline{1,l_1} \quad (3)$$

$$f(k_v,k_{v-1})=f(k_v,k_{v-1}),\ldots,f(k_{v+p2},k_{v+p2-1}),\ldots,f(k_{vl_2+1},k_{vl_2}),\ p_2=\overline{1,l_2}\ . \quad (4)$$

Functions of intercut connectivity $f(k_{v+p_1-1},k_{v+p_1})$ and $f(k_{v+p_2},k_{v+p_2-1})$ can be written as follows:

$$f(k_{v+p_1-1},k_{v+p_1})=\sum_{j=0}^{M-1}\sum_{i=0}^{N-1}\left(a_{ij}^{k_{v+p1-1}}\wedge a_{ij+1}^{k_{v+p1}}+a_{ij}^{k_{v+p1-1}}\wedge a_{ij-1}^{k_{v+p1}}+a_{ij}^{k_{v+p1-1}}\wedge a_{i-1j}^{k_{v+p1}}+\ldots+a_{ij}^{k_{v+p1-1}}\wedge a_{i+j+1}^{k_{v+p1}}\right). \quad (5)$$

Similarly, you can write the intercut connectivity function

$$f(k_{v+p2},k_{v+p2-1})=\sum_{j=0}^{M-1}\sum_{i=0}^{N-1}\left(a_{ij}^{k_{v+p2}}\wedge a_{ij+1}^{k_{v+p2-1}}+a_{ij}^{k_{v+p2}}\wedge a_{ij-1}^{k_{v+p2-1}}+a_{ij}^{k_{v+p2}}\wedge a_{i-1j}^{k_{v+p2-1}}+\ldots+a_{ij}^{k_{v+p2}}\wedge a_{i+1j+1}^{k_{v+p2-1}}\right). \quad (6)$$

The general segmentation scheme can be represented by sequential or parallel definition of sequences of intercut connectivity functions:

$$f(k_v,k_{v+1}),f(k_{v+2},k_{v+3}),\ldots,f(k_{w-1},k_w) \quad (7)$$

and

$$f(k_v,k_{v-1}),f(k_{v-2},k_{v-3}),\ldots,f(k_2,k_1). \quad (8)$$

The determination of the intercut connectivity functions in accordance with formulas (7) and (8) is carried out until

$$f(k_v,k_{v+1})=\delta,\ f(k_{v+2},k_{v+3})=\delta,\ f(k_{w-1},k_w)=\delta\ , \quad (9)$$

or

$$f(k_v,k_{v-1})=\delta,\ f(k_{v-2},k_{v-3})=\delta,\ f(k_2,k_1)=\delta\ . \quad (10)$$

Practically, the threshold δ in expressions (9), (10) is formed for each function of intercut connectivity $f(k_{v+p_1+1}, k_{v+p_1})$ and $f(k_{v+p_2+1}, k_{v+p_2})$. The process of determining the functions of intercut connectivity in accordance with expressions (3) and (4) is carried out until $f(k_{v+p_1+1}, k_{v+p_1}) = \delta$ or $f(k_{v+p_2+1}, k_{v+p_2}) = \delta$. Moreover, in the roughest approximation, the elements of the cuts k_{v+p_1}, k_{v+p_2} for which the functions of intercut connectivity $f(k_{v+p_1+1}, k_{v+p_1}) = \delta$ and $f(k_{v+p_2+1}, k_{v+p_2}) = \delta$, respectively, are uninformative elements and should be excluded from the original image.

If $f(k_{v+p_1+1}, k_{v+p_1}) \neq \delta$ or $f(k_{v+p_2+1}, k_{v+p_2}) \neq \delta$, then iterative determination of the intercut connectivity functions is performed in accordance with expressions (3) and (4).

Let's consider an example for a particular case. Let *v=2*, then for the cut k_2, the intercut connectivity functions take the following form:

$$f(k_2, k_{2+1}) = f(k_2, k_{2+1}), f(k_{2+1}, k_3), \ldots, f(k_{2+p_1-1}, k_{2+p_1}), \ldots, f(k_{l_1-1}, k_{l_1}) \quad (11)$$

$$f(k_2, k_{2-1}) = f(k_2, k_{2-1}), f(k_{2-1}, k_{2-2}), \ldots, f(k_{2+p_2}, k_{2+p_2-1}), \ldots, f(k_{l_2+1}, k_{l_2}). \quad (12)$$

At the same time, for expressions (10) and (11), we sequentially define the functions of intercut connectivity: from $f(k_2, k_{2+1})$ to $f(k_{l_1-1}, k_{l_1})$ and from $f(k_2, k_{2-1})$ to $f(k_{l_2+1}, k_{l_2})$.

Let's assume that

$$f(k_2, k_{2+1}) \neq \delta, f(k_{l_1-2}, k_{l_1-1}) \neq \delta, \text{ and } f(k_{l_1-1}, k_{l_1}) = \delta$$

$$f(k_2, k_{2-1}) \neq \delta, .. f(k_{l_2+2}, k_{l_2+1}) \neq \delta \text{ and } f(k_{l_2+1}, k_{l_2}) = \delta. \quad (13)$$

This means that in the cuts k_{l_1} and k_{l_2}, some elements are uninformative. There are two possible solutions. The first is to directly remove the noise elements from the cuts. The second option is the most promising. When implementing it, it should be assumed that the cuts k_{l_1} and k_{l_2} define noise elements as the product of single elements by the value of an element, for example, selected by the average value. Thus, it is not the values of single elements themselves that are subtracted from the original image (image fragment), but the single elements of the cuts k_{l_1} and k_{l_2} multiplied by the corresponding values of the minimum elements for this cut.

Thus, for all cuts with the maximum value of the internal cut connectivity function between cuts k_{1l_1} and k_{2l_2}, k_{2l_2} and k_{1l_1}, k_{2l_1} and k_{3l_1}, k_{3l_1} and k_{2l_1}, as well as between cuts k_{1l_2} and its preceding cut and k_{3l_2} and its following cut, and vice versa for the last two pairs of cuts. For the sake of clarity, let us consider the definitions of the intercut connectivity functions for the cuts k_{2l_2}, k_{1l_1} and k_{1l_1}, k_{2l_2}.

$$f(k_{2l_2}, k_{1l_1}) = f(k_{2l_2}, k_{2l_2+1}), f(k_{2l_2+1}, k_{2l_2+2}), .., f(k_{1l_1+1}, k_{1l_1})$$

$$f(k_{1l_1}, k_{2l_2}) = f(k_{1l_1}, k_{1l_1+1}), f(k_{1l_1+1}, k_{1l_1+2}), .., f(k_{2l_2-1}, k_{2l_2}). \quad (14)$$

If the elements in the cuts k_{2l_2} and k_{1l_1} are uninformative, then for the following intercut connectivity functions, the conditions for determining whether the elements of the cut are informative or not change the meaning. That is, the expressions

$$f\left(k_{2l_2}, k_{2l_2+1}\right) \neq \delta,\ f\left(k_{2l_2+1}, k_{2l_2+2}\right) \neq \delta, \ldots, f\left(k_{1l_1+1}, k_{1l_1}\right) \neq \delta$$

or

$$f\left(k_{1l_1}, k_{1l_1+1}\right) \neq \delta,\ f\left(k_{1l_1+1}, k_{1l_1+2}\right) \neq \delta, \ldots, f\left(k_{2l_2-1}, k_{2l_2}\right) \neq \delta \qquad (15)$$

imply, that the elements of the cuts $k_{2l_2+1}, k_{2l_2+2}, .., k_{1l_1}$ are uninformative. Therefore, for them, the average value of the elements should be subtracted from the elements of the original image. Moreover, for a rough approximation, the elements of the cuts located between the cuts k_{1l_1} and k_{2l_2} are uninformative and the latter are excluded from the original image. If the elements of the cut k_{2l_2} are uninformative, the unit values of this cut are transferred to the next cut k_{2l_2-1} and the intercut connectivity function is determined for this (changed) cut. Further, this iterative process of transferring unit elements of a cut can be continued. If condition (15) is not met, the elements of the cut under consideration are considered image elements.

If we assume that all the elements belonging to an image are interconnected, i.e. correlated in the space-time domain, the following problem arises: how can we optimally implement the multi-stage segmentation operation, assuming that the image elements are correlated in space and time, and the non-informative elements are decorrelated.

The k-means clustering algorithm in pattern recognition tasks. Extracting information from images and understanding it in such a way that the obtained information can be used to perform various tasks is a key aspect of machine learning. Image segmentation is one of the first steps towards understanding images and subsequently identifying different objects within them. Image segmentation involves dividing or partitioning an image into regions with similar attributes. It is one of the most critical components of image analysis and pattern recognition, and still remains one of the most challenging tasks in image processing and analysis. It has applications in multiple fields such as remote image analysis, medicine, traffic monitoring, fingerprint recognition, and more [23-26]. Segmentation methods involving classification approaches face significant challenges in computing the number of clusters present in the object space or in extracting the relevant function. This type of image segmentation is widely used due to its ease of understanding and relatively high accuracy of results. There are several methods, and one of the most popular is the k-means clustering algorithm. This clustering algorithm is unsupervised and is used to segment the region of interest from the background.

When using clustering algorithms, the task arises of determining the number of cluster centers for further analysis. For example, when clustering using the k-means algorithm, which is an exclusive clustering method, the number of centers is set initially and the algorithm works recursively until the formed clusters are balanced according to the proximity of individual data elements to the cluster centre. However, if the data set obtained as a result of experimental research is large enough, it is impossible to determine the optimal number of centers. If a certain number of centers is specified, the clusters formed using the k-means method will not correspond to the internal structure of the data. Therefore, using this approach for recognition tasks will be ineffective. When using, for example, hierarchical clustering methods, the algorithm works until all data is assigned to a single cluster. And there's also the question of which clustering level to stop at so that the number of clusters you get is the best for further analysis.

The paper looked at different ways to pick the best number of clusters, but the most effective one was the one that considered the max density of data assigned to each cluster at different clustering levels. The density was calculated as the ratio of data elements included in the cluster to the volume of this cluster. This approach allows for the structural grouping of data elements and reduces the running time of the clustering algorithm, since it will not be processed to the end (for hierarchical clustering methods).

Detection of overheated axleboxes. As previously noted, the railway industry faces the problem of monitoring the technical condition of axleboxes for safety hazards. Therefore, we propose using a machine vision system to monitor the thermal characteristics of moving train components based on thermographic cameras and the unsupervised k-means clustering algorithm. For modeling purposes, two images of axleboxes were used: a normal axlebox (Fig. 4(a)) and an overheated axlebox (Fig. 4(b)).

Note on images: since it is not possible to take real images of axle boxes in the infrared range due to the high cost of thermographic cameras, a decision was made to convert ordinary RGB images into a form that visually creates the effect that the image is taken in the thermal range. Therefore, the image of the axle box, which we have conditionally labeled as normal, could actually also be overheated - and, unfortunately, this factor distorts the results of the modeling.

Thus, for modeling, color images of both normal and overheated axle boxes are used for defective segmentation of thermal indicators. Defective segmentation is carried out in two stages. First, pixels are clustered based on their color and spatial characteristics, where the clustering process is performed. Then, the clustered blocks are combined into a certain number of regions. By using this two-step procedure, computational efficiency can be increased by avoiding feature extraction for each pixel in the axle box images. Although color is typically not used for defect segmentation, it provides high discriminative power for different areas of the image. Therefore, this approach provides a reliable solution for segmenting thermal indicators.

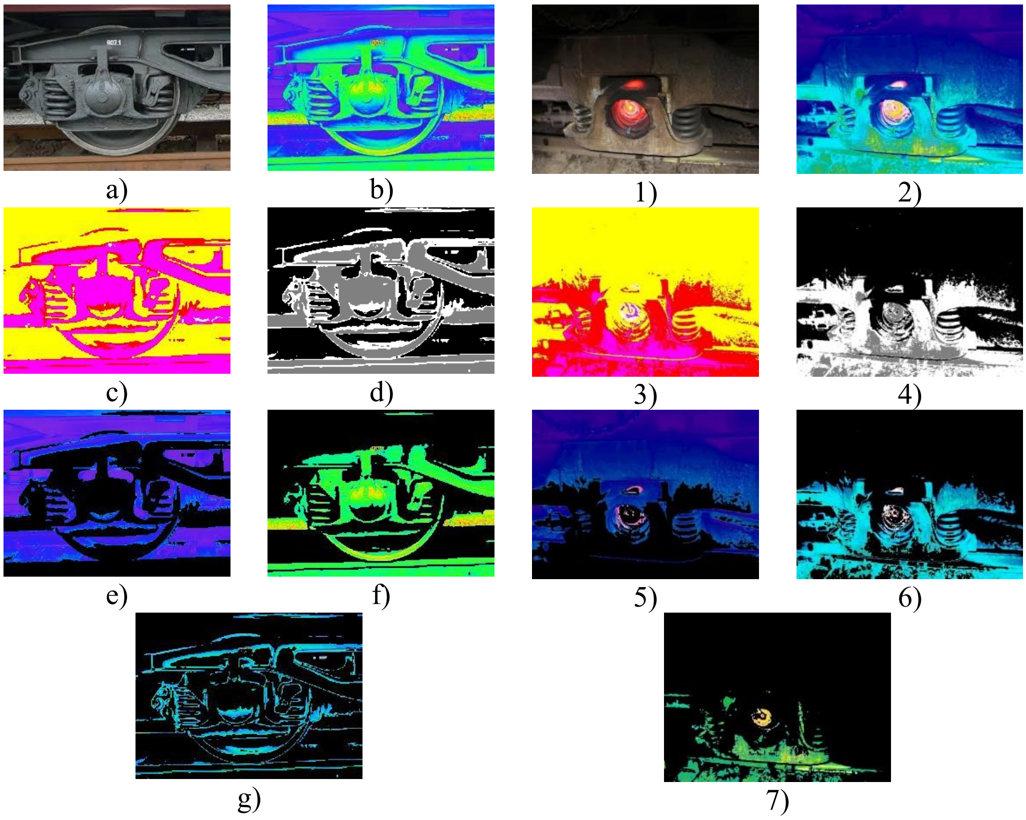


Fig. 4. **Segmentation based on color using the k-means clustering algorithm: a) and 1) - Conditional images of axle boxes; b) and 2) - Illustration of normal and overheated axle boxes in the infrared range; c) and 3) - Thermal images converted from the RGB color space to L*a*b* color space; d) and 4) - Images labeled with cluster indices; e) and 5) - Objects in cluster 1; f) and 6) - Objects in cluster 2; g) and 7) - Objects in cluster 3.**

Experimental results show the effectiveness of the proposed approach for improving the quality of thermal image segmentation in terms of accuracy and computational time. The results of the modeling demonstrate that the proposed approach is promising. Segmentation using the k-means algorithm is very useful for image analysis. An important goal of image segmentation is to separate the object and background, regardless of whether the image has a blurred boundary. Defective segmentation of thermal characteristics can be considered as an example of image segmentation where the number of segments is unknown. The main goal of the proposed approach is automatic color segmentation using the k-means clustering method and the L*a*b* color space [27-29].

The introduced segmentation structure works in five steps as follows:

1. Reading images in the infrared range.
2. Conversion of the image from RGB color space to L*a*b*. The L*a*b* color space is used because it consists of a luminance layer in the "L" channel and two chromaticity layers in the "a*" and "b*" channels. The use of the L*a*b* color space is uncompromising because all color information is present in "a*" and "b*".
3. Color classification using the k-means clustering algorithm in the "a*b*" space. To measure the difference between two colors, the Euclidean distance metric is used.
4. Labeling each pixel in the image using the results of the k-means algorithm. For each pixel in our input, the algorithm calculates an index corresponding to the cluster. Each pixel in the image will be labeled with the cluster index.
5. Creation of images that segment the input image by color. We need to separate the pixels in the image by color using pixel labels, which will result in different images based on the number of clusters. We program the identification of the index of each cluster that contains the highest temperature readings because the algorithm does not return the same cluster index every time. However, we can do this by using the central value of the clusters, which contains the average "a*" and "b*" values for each cluster.

Conclusions. The k-means clustering algorithm is classified as an unsupervised data clustering algorithm (unsupervised learning). The advantages of this algorithm include its clarity, simplicity, and speed of use. However, its drawbacks include slow performance with large datasets and sensitivity to outliers, which can distort the mean value. The simulation results showed that the k-means algorithm can effectively describe relatively simple shapes. In addition, this algorithm assumes that all clusters have the same ‘diameter’ in a certain sense, it always draws the boundary between clusters so that it passes exactly in the middle between the centres of the clusters.

A scheme for implementing a machine vision system for detecting overheated axle boxes in railway transport using the k-means clustering algorithm has been proposed.

Experimental results of the modeling showed that segmentation based on the color model using the k-means clustering algorithm allows for accurate segmentation of temperature indicators of rolling stock. For example, the image in Fig. 5(e) shows the segmentation of objects in the second cluster and, in comparison with the conditional thermal image (Fig. 5(b)), represents the recognition of the dominant green color, indicating a satisfactory temperature condition of the axle box. This is not the case with the image of the overheated axle box (Fig. 5(6)), where the white color signals an excessive temperature increase, indicating an emergency condition of the axle box.

It should be noted that the application of this algorithm in the development of software for thermographic cameras aimed at recognizing temperature indicators is effective, as this algorithm allows for image compression, which will increase data processing speed and enable monitoring of the thermal characteristics of rolling stock at higher speeds.

REFERENCES

1. Ghaziasgar, M., Bagula, A., & Thron, C. (2020). Computer Vision Algorithms for Image Segmentation, Motion Detection, and Classification. *Implementations and Applications of Machine Learning. Studies in Computational Intelligence*, *782*, 119-138. Springer, Cham. https://doi.org/10.1007/978-3-030-37830-1_5.
2. Hassner, T., & Liu, C. (Eds.). (2016). *Dense image correspondences for computer vision* (pp. 3-14). Springer International Publishing. https://doi.org/10.1007/978-3-319-23048-1.
3. Vishwakarma, A., Dasgupta, A., & Racherla, V. (2025). Detecting and classifying multiple track defects using clustering algorithm. *Engineering Structures*, *334*, 120197. https://doi.org/10.1016/j.engstruct.2025.120197.
4. Pacheco, F., Cerrada, M., Sánchez, R. V., Cabrera, D., Li, C., & de Oliveira, J. V. (2017). Attribute clustering using rough set theory for feature selection in fault severity classification of rotating machinery. *Expert Systems with Applications*, *71*, 69-86. https://doi.org/10.1016/j.eswa.2016.11.024.

5. Jiang, Y., Wang, H., Tian, G., Yi, Q., Zhao, J., & Zhen, K. (2019). Fast classification for rail defect depths using a hybrid intelligent method. *Optik*, *180*, 455-468. https://doi.org/10.1016/j.ijleo.2018.11.053.
6. Rapp, S., Martin, U., Strähle, M., & Scheffbuch, M. (2019). Track-vehicle scale model for evaluating local track defects detection methods. *Transportation Geotechnics*, *19*, 9-18. https://doi.org/10.1016/j.trgeo.2019.01.001.
7. Li, Q., Yao, K., & Zhang, X. (2020). A change-point detection and clustering method in the recurrent-event context. *Journal of Statistical Computation and Simulation*, *90*(6), 1131–1149. https://doi.org/10.1080/00949655.2020.1718149.
8. Gomez, M. J., Castejon, C., Corral, E., & Cocconcelli, M. (2023). Railway axle early fatigue crack detection through condition monitoring techniques. *Sensors*, *23*(13), 6143. https://doi.org/10.3390/s23136143.
9. Amini, A., Entezami, M., Huang, Z., Rowshandel, H., & Papaelias, M. (2016). Wayside detection of faults in railway axle bearings using time spectral kurtosis analysis on high-frequency acoustic emission signals. *Advances in Mechanical Engineering*, *8*(11), 1687814016676000. https://doi.org/10.1177/1687814016676.
10. Lisanti, G., Karaman, S., Pezzatini, D., & Bimbo, A. D. (2018). A multi-camera image processing and visualization system for train safety assessment. *Multimedia Tools and Applications*, *77*, 1583-1604. https://doi.org/10.1007/s11042-017-4351-4.
11. Wei, X., Yang, Z., Liu, Y., Wei, D., Jia, L., & Li, Y. (2019). Railway track fastener defect detection based on image processing and deep learning techniques: A comparative study. *Engineering Applications of Artificial Intelligence*, *80*, 66-81. https://doi.org/10.1016/j.engappai.2019.01.008.
12. Jamshidi, A., Roohi, S. F., Núñez, A., Babuska, R., De Schutter, B., Dollevoet, R., & Li, Z. (2016). Probabilistic defect-based risk assessment approach for rail failures in railway infrastructure. *IFAC-PapersOnLine*, *49*(3), 73-77. https://doi.org/10.1016/j.ifacol.2016.07.013.
13. Yiakopoulos, C. T., Gryllias, K. C., & Antoniadis, I. A. (2011). Rolling element bearing fault detection in industrial environments based on a K-means clustering approach. *Expert Systems with Applications*, *38*(3), 2888-2911. https://doi.org/10.1016/j.eswa.2010.08.083.
14. Botvin, M., Butryk, N., & Gertsiy, O. (2019). K-means clustering algorithm in image recognition tasks. *Collection of scientific works of the State University of Infrastructure and Technologies series "Transport Systems and Technologies"* (34), 199–209. https://doi.org/10.32703/2617-9040-2019-34-2-3.
15. Zhang, C., Zhong, P., Liu, M., Song, Q., Liang, Z., & Wang, X. (2022). Hybrid Metric K-Nearest Neighbor Algorithm and Applications. *Mathematical Problems in Engineering*, *2022*(1), 8212546. https://doi.org/10.1155/2022/8212546.
16. Ando, K., Kuniyoshi, Y., Onogi, N., Uchitane, T., Mukai, N., Iwata, K., Ito, N., & Jiang, Y. (2024). Cluster Detection for Traffic Accidents on Spatiotemporal Networks. *Procedia Computer Science*, *246*, 371-380.. https://doi.org/10.1016/j.procs.2024.09.416.
17. Huang, B., Zhu, Y., Wang, Z., & Fang, Z. (2021). Imbalanced data classification algorithm based on clustering and SVM. *Journal of Circuits, Systems and Computers*, *30*(02), 2150036. https://doi.org/10.1142/S0218126621500365.
18. Timchenko, L. I., Basov, V. I., Gertsiy, O. A., Kokriatskaia, N. I., & Ivasiyk, I.D., (2010). Rospiznavannya pivtonovych zobrazhen za shemoyu "gruboi" - "tochnoi" obrobki: Monografia. [Recognition of halftone images according to the scheme of "rough" - "accurate" processing: Monograph]. *Kyiv: Naukova Dymka,* 180 p. ISBN 978-966-00-0904-6.
19. Timchenko, L. I., Kokriatskaia, N. I., Gertsiy, A. A., Stepaniuk, D. S., Chernyaschuk, N. L., Kotyra, A., & Amirgaliyev, Y. (2019, November). Elaboration of pyramidal methods applying computation technique "rough-fine" image identification. In *Photonics Applications in Astronomy, Communications, Industry, and High-Energy Physics Experiments 2019* (Vol. 11176, pp. 384-391). SPIE. https://doi.org/10.1117/12.2537179.
20. Orazayeva, A., Wójcik, W., Pavlov, S., Tymchenko, L., Kokriatska, N., Tverdomed, V., ... & Semenova, L. (2022, December). Biomedical image segmentation method based on contour preparation. In *Photonics Applications in Astronomy, Communications, Industry, and High Energy Physics Experiments 2022* (Vol. 12476, pp. 21-26). SPIE. https://doi.org/10.1117/12.2657929.
21. Timchenko, L., Kokriatskaya, N., Tverdomed, V., Stetsenko, O., Kaplun, V., Kolesnytskyj, O. K., ... & Zhunissova, U. (2023). Segmentation of multigradation images based on spatial connectivity features. *Informatyka, Automatyka, Pomiary w Gospodarce i Ochronie Środowiska*, *13*(3), 47-50. https://doi.org/10.35784/iapgos.5352.
22. Fan, Z., Xie, J. K., Wang, Z. Y., Liu, P. C., Qu, S. J., & Huo, L. (2021, May). Image classification method based on improved KNN algorithm. In *Journal of physics: Conference series* (Vol. 1930, No. 1, p. 012009). IOP Publishing. https://doi.org/10.1088/1742-6596/1930/1/012009.
23. Gertsiy, O. (2023). Models of criterion evaluation of the image processing systems effectiveness. *Collection of scientific works of the State University of Infrastructure and Technologies series "Transport Systems and Technologies"*, (41), 143-154. https://doi.org/10.32703/2617-9059-2023-41-12.
24. Gertsiy, O. (2024). Research on graphic data formats for compact representation and comparison of images. *Collection of scientific works of the State University of Infrastructure and Technologies series "Transport Systems and Technologies"*, (43), 173–187. https://doi.org/10.32703/2617-9059-2024-43-14.
25. Sacerdoti, F. M., Giordano, A., & Cavaliere, C. (Eds.). (2016). *Advanced imaging techniques in clinical pathology* (No. 9210). Springer New York. https://doi.org/10.1007/978-1-4939-3469-0_2.
26. Burger, W., & Burge, M. J. (2022). *Digital image processing: An algorithmic introduction*. Springer Nature. https://doi.org/10.1007/978-3-031-05744-1.
27. Sinaga, A. S. R. (2019). Color-based Segmentation of Batik Using the L*a*b Color Space. *Sinkron: Jurnal Dan Penelitian Teknik Informatika*, *3*(2), 175-179. https://doi.org/10.33395/sinkron.v3i2.10102.

28. Abbadi, N. K. E., & Razaq, E. S. (2020). Automatic gray images colorization based on lab color space. *Indonesian Journal of Electrical Engineering and Computer Science*, *18*(3), 1501-1509. https://doi.org/10.11591/ijeecs.v18.i3.pp1501-1509.
29. Saifullah, S., Prasetyo, D. B., Dreżewski, R., & Dwiyanto, F. A. (2023). Palm oil maturity classification using K-nearest neighbors based on RGB and L* a* b color extraction. *Procedia Computer Science*, *225*, 3011-3020. https://doi.org/10.1016/j.procs.2023.10.294.

Олександр Герцій[1], Сергій Карнатов[2], Віталій Гладиш[3], Валентина Ткаченко[4]

[1]Доцент, Кафедра Автоматизація та комп'ютерно-інтегровані технології транспорту, Національний транспортний університет, вул. Михайла Омеляновича-Павленка, 2, 01010, м. Київ, Україна. ORCID: https://orcid.org/0000-0002-8634-5880.
[2]Аспірант, Кафедра Автоматизація та комп'ютерно-інтегровані технології транспорту, Національний транспортний університет, вул. Михайла Омеляновича-Павленка, 2, 01010, м. Київ, Україна. ORCID: https://orcid.org/0009-0006-6254-6166.
[3]Аспірант, Кафедра Автоматизація та комп'ютерно-інтегровані технології транспорту, Національний транспортний університет, вул. Михайла Омеляновича-Павленка, 2, 01010, м. Київ, Україна. ORCID: https://orcid.org/0009-0003-5801-056X.
[4]Доцент, Кафедра транспортного права та логістики, Національний транспортний університет, вул. Михайла Омеляновича-Павленка, 2, 01010, м. Київ, Україна. ORCID: https://orcid.org/0000-0001-5147-0772.

Моделювання алгоритму кластеризації зображень для виявлення перегрітих букс вагонів

В статті розглянуто ідентифікацію зображень на основі алгоритму кластеризації для виявлення теплових характеристик букс вагонів. Проведено аналіз перспектив застосування алгоритмів кластерізації в задачах ідентифікації цифрових зображень. Наведено принцип роботи запропонованого алгоритму при сегментації та стисненні графічних зображень, а також при розпізнаванні образів у транспортній сфері. Представлено блок-схему алгоритму і описано принцип його роботи. Пропонується схема реалізації системи машинного зору з використанням даного алгоритму в процесі виявлення перегрітих букс вагонів. Проведено експериментальне моделювання по сегментації на основі кольорових моделей зображень поданих в інфрачервоному діапазоні для виявлення ділянок з підвищеною температурою засобами програми MATLAB. Для експерименту обрані зображення, на яких представлено букси рухомого складу, одна з яких нормальна, а друга - перегріта. В результаті проведеного моделювання виявлено, що сегментація на основі кольорових моделей зображень дозволяє точно сегментувати термозображення букси. Дослідження показали, що застосування запропонованого алгоритму і його програмна реалізація для термографічних камер є ефективним з метою розпізнавання температурних показників. Даний алгоритм також дозволяє представити зображення в компактному вигляді, що приводить до підвищення швидкості його обробки і дозволяє швидко проводити моніторинг теплових характеристик букс рухомого складу.

Ключові слова: *термографічні зображення, кластеризація, сегментація, моніторинг, моделювання, температурні показники, букса.*