SILESIAN UNIVERSITY OF TECHNOLOGY PUBLISHING HOUSE

SCIENTIFIC PAPERS OF SILESIAN UNIVERSITY OF TECHNOLOGY 2025
ORGANIZATION AND MANAGEMENT SERIES NO. 234

APPLICATION OF GAS CHROMATOGRAPHY METHODS IN INDUSTRIAL IMAGE RECOGNITION

Mariusz ŚWIĘCICKI

Cracow University of Technology; mariusz.swiecicki@pk.edu.pl, ORCID: 0000-0002-5929-4619

Purpose: The main purpose of the study is to develop and test an image classification algorithm inspired by the mechanisms of gas chromatography. The research aims to transfer theoretical concepts from analytical chemistry—such as retention time and the number of theoretical plates—into the field of image processing. The motivation stems from the need for effective extraction and differentiation of visual features, particularly in domains where internal structural differences are subtle or where shape deformation occurs.
Design/methodology/approach: The proposed method treats input images as complex mixtures of visual components (analogous to chemical substances) and decomposes them into subimages that are transformed into chromatograms. The algorithm performs feature extraction using simulated chromatographic separation, followed by background subtraction and peak comparison. The classification is based on matching the extracted chromatographic peaks against reference classes. Two types of datasets were used to validate the approach: synthetic blood cell images (medical domain) and geometric shape figures (industrial domain), both with controlled distortions. The influence of subimage size and the number of theoretical plates on classification performance was systematically tested.
Findings: The experiments confirm that the classifier's effectiveness strongly depends on both the resolution of the chromatographic decomposition (i.e., the number of plates) and the subimage size. For blood cells, the best results (precision 0.78) were achieved with a subimage size of 4×4 and a high number of plates (100,000). In the case of shape classification, the highest precision (0.84) was obtained for 60×60 subimages and 50 plates. The study shows that excessive resolution can lead to overfitting, while too little generalization limits the algorithm's sensitivity to subtle differences.Research limitations/implications: The approach is tested on synthetic datasets, which—while offering control over feature variance—may not reflect all the complexities of real-world data. Future research should include validation on real medical images and industrial visual inspection systems. Additionally, comparative benchmarks against conventional machine learning classifiers (e.g., CNNs) could strengthen the results.
Practical implications: The proposed method is applicable in areas requiring robust and explainable classification of complex image data. In medicine, it can support blood diagnostics by highlighting morphologically relevant features of leukocytes. In industry, it can enhance automated inspection systems by enabling recognition of deformed parts without reliance on purely edge-based or shape-based detection.
Originality/value: The article presents an original and interdisciplinary algorithm that combines principles of gas chromatography with image analysis. It introduces a novel metaphor for feature decomposition and demonstrates its applicability in both medical and industrial

scenarios. This approach expands the set of tools available for interpretable and structure-aware image classification.
Keywords: gas chromatography, image classification, feature extraction, synthetic dataset, biomedical imaging.
Article classification: Research paper, Technical paper.

1. Introduction

Image recognition is a key area of research in artificial intelligence, image processing, and computer vision. The main goal of this process is the identification and classification of objects, patterns, or scenes depicted in digital images. Systems designed for image recognition face numerous challenges, such as varying lighting conditions, noise, rotation, scaling, geometric deformations, and partial occlusion of the objects being analyzed.

Modern approaches to image recognition are primarily based on machine learning techniques, especially deep learning methods that enable effective extraction and representation of relevant visual features. These techniques, including convolutional neural networks (CNNs), allow for automatic learning of hierarchical representations from large datasets. In contexts where interpretability is crucial, classical feature extraction methods such as HOG (Histogram of Oriented Gradients) or SIFT (Scale-Invariant Feature Transform) are also applied.

This study introduces an innovative image recognition approach inspired by the mechanisms of gas chromatography. The proposed method treats image data as mixtures of visual components that undergo a separation process based on their structural, textural, or color properties. A key element of the method is the analogy to chromatographic processing—each image fragment is assigned a so-called retention time, which reflects its “affinity” toward a stationary phase in a modeled chromatographic column.

This decomposition enables the extraction of features essential for classification while reducing the influence of interfering components. By applying chromatographic separation principles, the image is transformed into a chromatogram, which serves as a compact representation suitable for further analysis, segmentation, or classification.

The proposed method can serve as an alternative or a complement to classical image processing techniques, especially in applications requiring interpretability, robustness to deformation, and fine-grained differentiation of internal object structures.

2. Chromatographic Data Processing Method

The chromatographic data decomposition method, inspired by the principles of gas chromatography, is based on specific theoretical assumptions. It is assumed that input vectors submitted for classification represent mixtures of components with an unknown composition.

In the proposed algorithm, it is assumed that the decomposition of the input vector results in the formation of sub-vectors of equal length, which enables their further analytical processing. This approach mirrors the separation process observed in classical chromatography but is adapted to the context of multidimensional data representation and analysis.

2.1. Principle of chromatographic data processing

The core idea of the chromatographic data decomposition method is as follows: the method consists of several processing stages. In the first stage, the input vector is divided into smaller sub-vectors, for which the affinity with respect to the stationary phase is calculated. This parameter determines the migration speed of individual sub-vectors through a modeled chromatographic column, directly affecting the so-called retention time—that is, the duration a given sub-vector remains within the column.

After the migration is completed, the vectors exiting the column are counted. Vectors with identical retention times are treated by the algorithm as equivalent in terms of data representation. As a result of this process, a chromatographic spectrum is obtained, which reflects the relationship between the number of vectors sharing the same retention time and their corresponding migration time.

The chromatographic decomposition algorithm has been described in detail in previous publications (Święcicki, 2024). Therefore, this article presents only those components of the algorithm that are directly relevant to the proposed image processing methods.

2.2. Image Representation

The image is represented as a two-dimensional array, where each element takes a value in the range from 0 to 255, corresponding to a grayscale representation. According to the fundamental assumption of the chromatographic data decomposition method, such an array is interpreted as a mixture of chemical compounds that are subject to further processing. In the next stage, the image is divided into subregions of equal size, treated as subarrays. Each of these subarrays is then processed by the algorithm, which computes a corresponding retention time for each one (Baxes, n.d.; Young et al., 1995).

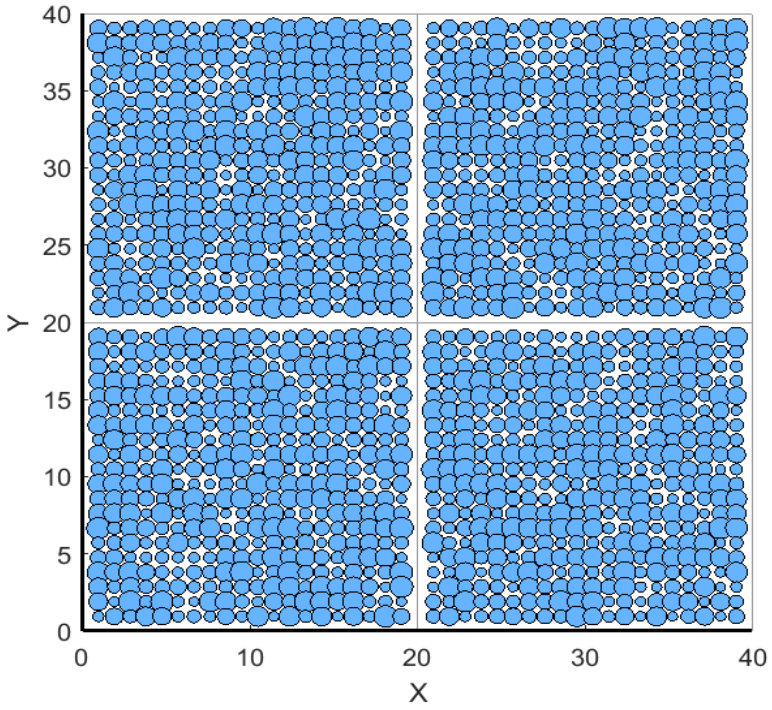

Figure 1. Division of the image, which is a mixture, into four subregions that will be treated as four "substances" for which the retention time will be calculated.

The **Błąd! Nie można odnaleźć źródła odwołania.** illustrates the process of converting an image into a chromatogram, which serves as the core structure used in subsequent stages of image processing. As demonstrated, the image—interpreted as a mixture—has been divided into one hundred subregions, each corresponding to an individual "substance" in the context of the chromatographic data decomposition algorithm. For each of these subregions, a retention time is calculated. It is important to note that the retention time value is directly dependent on the distribution of pixel values within the respective subregion.

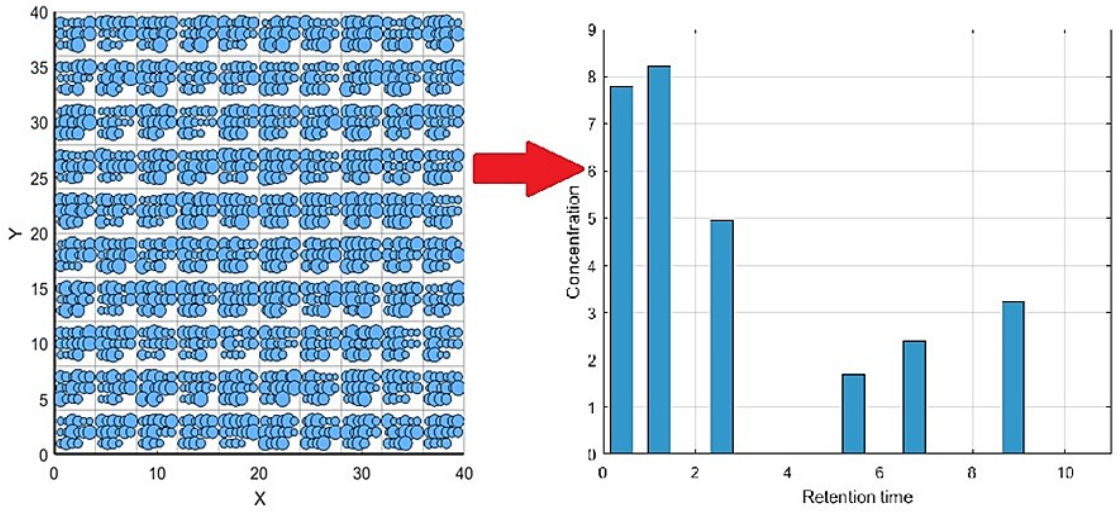

Figure 2. The process of converting an image into a chromatogram.

The result of calculating the retention time for each subregion and performing the grouping process is a chromatogram. The height of individual bars (peaks) in the chromatogram is determined by the number of subregions that share the same retention time.

2.3. Chromatographic Decomposition Algorithm

Algorithm 1 presents a method for chromatographic data separation inspired by the gas chromatography technique. The input data for the algorithm is an image P, which in a special case can be a color image represented as an RGB array. The next input parameters are N1 and M1, defining the size of the subregions into which the input image will be divided. The last parameter is a variable corresponding to the resolution in the chromatographic system, called the number of plates, determined by the length of the chromatographic column. The longer the chromatographic column, the higher the resolution of the chromatograph. Similarly, in the presented algorithm, increasing this parameter improves the resolution of the data separation.

Input: $P \in \mathbb{R}^{M \times N \times C}$; // Input image (mixture of substances)
Input: $M_1, N_1 \in \mathbb{N}$; // Dimensions of each sub-image representing a substance
Input: $n \in \mathbb{N}$; // Number of theoretical plates (discrete retention time levels)
Output: $C = \{(\tau_j, h_j)\} \subseteq \mathbb{N} \times \mathbb{N}$; // Chromatogram as retention times and peak heights

// Calculate the number of sub-images fitting into P without overlap

$$K = \left\lfloor \frac{M}{M_1} \right\rfloor \cdot \left\lfloor \frac{N}{N_1} \right\rfloor$$

// Extract sub-images

$$S = \{S_i\}_{i=1}^{K}, \quad S_i \subset P, \quad |S_i| = M_1 \times N_1$$

foreach $S_i \in S$ **do**
  // Calculate affinity to stationary phase for each substance
  $A_i \leftarrow \text{affinity}(S_i)$;
  // Compute discrete retention time

$$\tau_i \leftarrow \text{quantize}(f_{\text{retention}}(A_i), n) \in \{1, 2, \ldots, n\}$$

  // Measure signal intensity (e.g. average brightness)

$$\alpha_i \leftarrow \text{measure}(S_i)$$

// Group substances by retention time and sum intensities for chromatogram peaks

$$\forall \tau_j \in \{1, \ldots, n\}, \quad h_j = \sum_{\{i:\tau_i = \tau_j\}} \alpha_i$$

return *Chromatogram*

$$C = \{(\tau_j, h_j) \mid h_j > 0\}$$

Algorithm 1. Chromatogram-Based Data Decomposition Algorithm.

The algorithm begins by processing the input image P, which is represented as a two-dimensional matrix of dimensions $M \times N$. This image is treated as a mixture of substances, where each substance corresponds to a fragment of the image — a so-called subregion of size $M_1 \times N_1$. Dividing the image into such non-overlapping subregions allows for the analysis of local properties of the image assigned to individual substances.

For each subregion, an affinity value to the stationary phase is calculated, which is a key parameter in chromatography. Based on this value, the retention time is determined, describing how long a given substance "resides" in the chromatographic column. In the context of the algorithm, the retention time is discretized into one of the predefined points corresponding to chromatographic "plates," which ensures the resolution of substance recognition.

Next, substances (subregions) with the same retention time are grouped together. Such a group is represented as a single peak on the chromatogram, whose height is proportional to the number of substances belonging to that group. As a result, a histogram of retention times — a chromatogram — is created, which serves as the basis for further image analysis.

By applying this approach, the algorithm maps chromatographic properties, transferring the concept of mixture separation to the problem of image segmentation and classification. In this way, efficient recognition and grouping of image fragments with similar features is possible, which can support further image recognition tasks, such as classification or object detection.

3. Implementation of selected image processing methods

The image recognition process requires prior execution of appropriate preliminary operations on the image, which enable more effective extraction of features essential for further analysis. Therefore, in this work, selected image processing techniques are presented that have been applied as steps preceding the actual recognition process. These techniques have been implemented in the context of the developed chromatographic data separation algorithm, constituting an integral part of the input data preparation for analysis and classification.

In this section, the image processing operations carried out using mechanisms and data provided by the chromatographic data separation algorithm will be presented. The foundation of all presented operations is the chromatogram, which serves as a key structure representing information about the internal organization and composition of the processed image. It can be stated that the chromatogram functions as the central element in the image analysis process and is utilized in all the proposed algorithms.

Thanks to the direct linkage between a specific peak in the chromatogram and its corresponding subregion — treated as a substance in the mixture that the image represents — selective processing of chosen image fragments is possible. All the presented operations can be interpreted as filtering actions, analogous to those occurring in actual chromatographic systems.

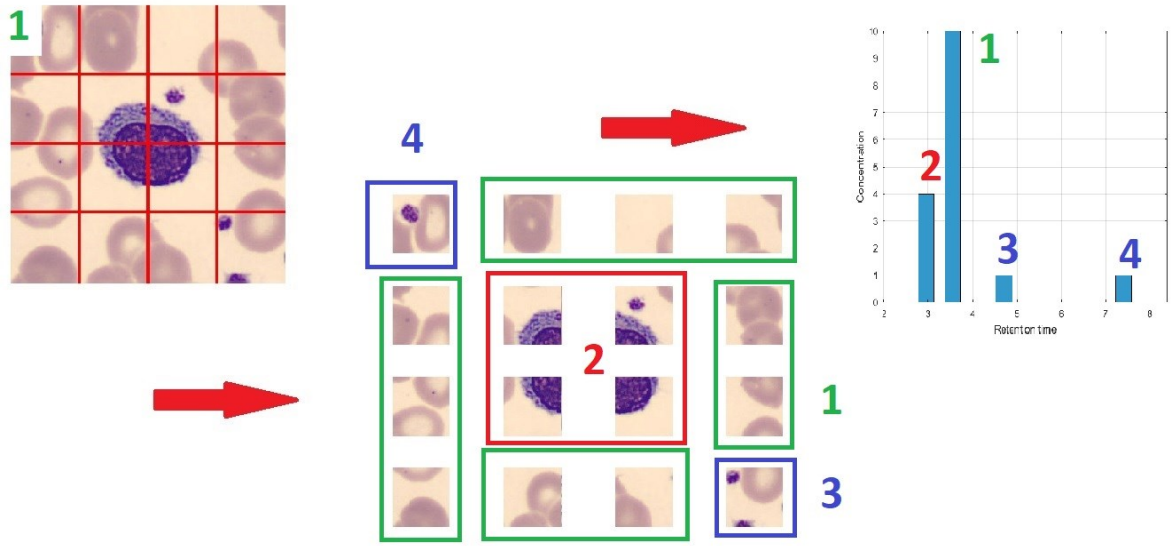

Figure 1. Phases of image processing using the chromatographic data separation algorithm.

Figure 1 illustrates the successive stages of the image processing procedure. In the first processing phase, the input image is divided into smaller subimages, which are interpreted as substances in a mixture. For each subimage, the level of interaction with the stationary phase is calculated, leading to the determination of its corresponding retention time.

In the next phase, grouping of substances—that is, subregions of the image—with similar or identical retention time values is performed. As shown in the presented diagram, multiple image fragments whose characteristics result in the same retention time can be assigned to a single peak in the histogram (chromatogram). The figure illustrates a case where different subimages with the same retention value are assigned to a peak marked with specific numbers, clearly indicating their similarity within the adopted analysis model(Young et al., 1995).

3.1. Image operations

In the following part of the article, the image preprocessing operations, which constitute a preparatory stage for the actual recognition process, will be presented. Although the general properties of these operations are well known and widely described in the literature, this article focuses on their implementation using the chromatographic data separation method (Young et al., 1995).

The image processing operations presented later in this work include two main groups. The first group comprises filtering operations based on chromatogram analysis. This category includes, among others, the chromatogram subtraction operation and the removal of common peaks, which enable selective extraction of significant image fragments. The second group consists of morphological operations, which are a key preparatory step in the image recognition process. Within this group, the implementation of the dilation operation (object expansion) will be demonstrated, allowing, among other things, the reconstruction of object structures and elimination of minor disturbances.

3.2. Removal of the peak with the highest concentration

The algorithm presents a sequence of operations aimed at eliminating selected substances from the mixture—represented by an image—based on chromatogram analysis. Specifically, those components (subimages) that exhibit the highest concentration, corresponding to the greatest number of occurrences of the same retention time value, are removed. The basis for the elimination decision is the structure of the chromatogram, which reflects the distribution of substances within the image. The input arguments of the algorithm are: the processed image, parameters defining the sizes of the subimages, and the number of chromatographic plates, which corresponds to the degree of resolution of the applied chromatographic data separation algorithm.

Input :
- $P \in \mathbb{R}^{M \times N}$: input image (mixture of substances)
- $M_1, N_1 \in \mathbb{N}$: dimensions of each subimage
- $R_{\min}, R_{\max} \in \mathbb{R}$: min and max retention time
- $P \in \mathbb{N}$: number of chromatographic plates

Output:
- C: Chromatogram as histogram map $C : \mathbb{R} \to \mathbb{N}$
- P_{out}: Output image with most concentrated retention time substances removed

Assumptions: The image P is partitioned into K non-overlapping subimages: $S = \{s_1, s_2, \ldots, s_K\}$ where $K = \left\lfloor \frac{M}{M_1} \right\rfloor \cdot \left\lfloor \frac{N}{N_1} \right\rfloor$, each $s_i \in \mathbb{R}^{M_1 \times N_1}$

Let $\Delta R = \frac{R_{\max} - R_{\min}}{P}$ `// Grid step of retention time`

Let $G = \{R_{\min} + j \cdot \Delta R \mid j = 0, \ldots, P\}$ `// Retention time grid`

Initialize map $RT : S \to G$, histogram $C : G \to \mathbb{N}$ with zero counts

foreach $s_i \in S$ **do**
- Compute raw retention time $RT_{\text{real}}(s_i)$
- Quantize: $RT_q(s_i) = \arg\min_{r \in G} |RT_{\text{real}}(s_i) - r|$
- Assign: $RT[s_i] \leftarrow RT_q(s_i)$
- Increment: $C[RT_q(s_i)] \leftarrow C[RT_q(s_i)] + 1$

end

Let $r^* = \arg\max_{r \in G} C[r]$ `// Most frequent retention time`

Define $S' = \{s_i \in S \mid RT[s_i] \neq r^*\}$ `// Remove most concentrated substances`

Reconstruct output image P_{out} from subimages in S'

return C, P_{out}

Algorithm 2. Algorithm for Eliminating Dominant Subimages from an Image.

The algorithm presented as the "Chromatographic elimination of the most abundant substances" is inspired by the process of gas chromatography and constitutes a part of image processing utilizing a chemical model. Its goal is to remove from the image those fragments (subregions) that occur most frequently — analogous to removing substances with the highest concentration from a mixture (Sparkman, 2005; Stilo et al., 2021).

The elimination algorithm for subregions with the highest concentration is based on the analogy to the gas chromatography process, in which a mixture of substances is separated in a column based on differences in retention times. In the first stage, the input image is treated as a set of non-overlapping subregions (substances), obtained by dividing the pixel matrix into blocks of specified dimensions. For each such subregion, an affinity value to the stationary phase is calculated — this parameter translates into a retention time, which is then discretized to the nearest point on a grid defined by the number of chromatographic plates.

In the next step, a chromatogram is created — a histogram reflecting the distribution of the number of subregions as a function of retention time. The retention time value for which the number of corresponding subregions is the highest indicates the dominant "substance" in the analyzed mixture. Then, all subregions with that retention time value are removed from the image, which corresponds to eliminating the component with the highest concentration in classical chromatography.

The result of the algorithm's operation is a new image, devoid of the most strongly represented fragments, and an updated chromatogram. This enables targeted removal of overrepresented features, improving the efficiency of subsequent stages of image recognition or segmentation. Thus, the algorithm serves as a useful preprocessing tool, reducing interference caused by dominant structures.

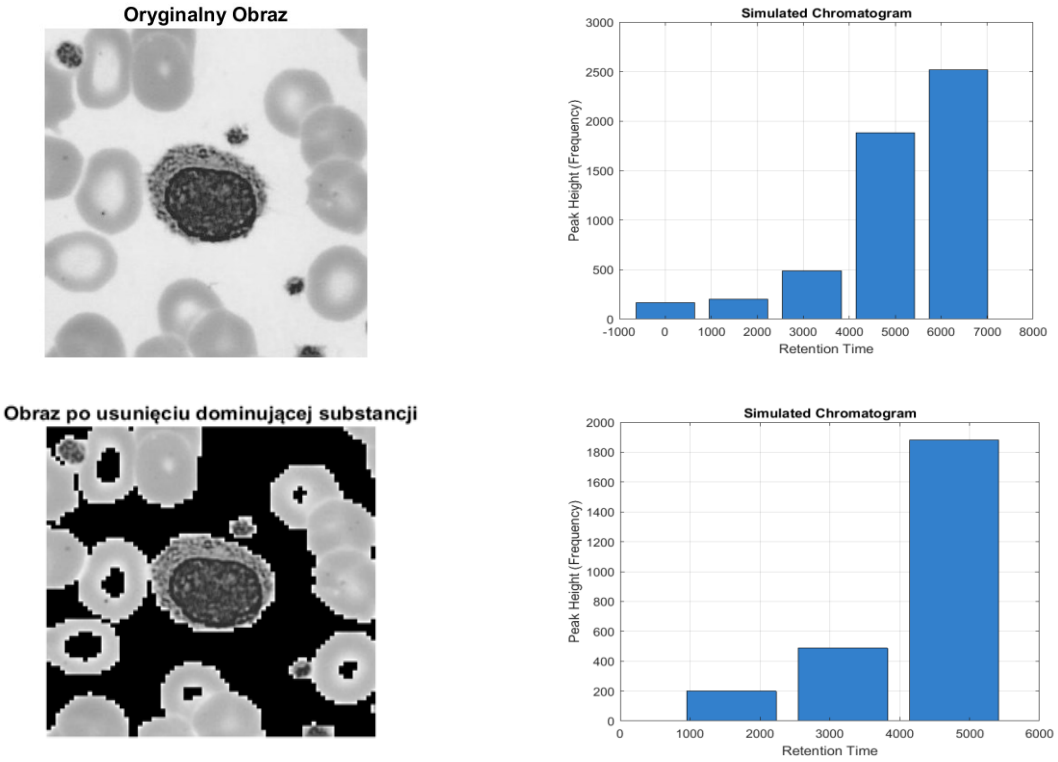

Figure 2. Image and chromatogram before and after removal of the dominant substance (number of plates = 4, sub- image size 5×5).

Figure 2 presents the effect of the algorithm for eliminating the substances with the highest concentration. As shown in the illustrations, the image at the bottom has had the fragment corresponding to the background removed. The second column contains chromatograms: on the left for the original image, and on the right for the image processed by the described algorithm.

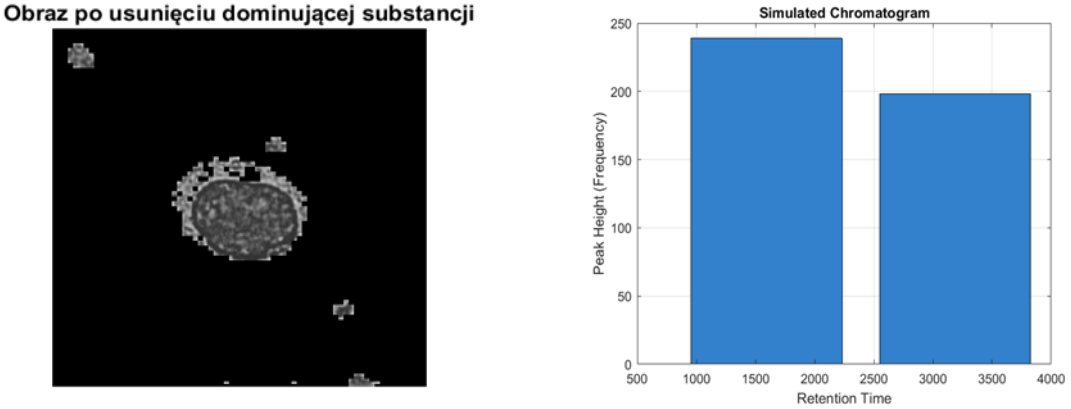

Figure 3. Image and chromatogram after removal of the dominant substance (number of plates = 4, sub-image size 5×5), obtained by three successive applications of the dominant-peak removal operation to the original image.

Figure 6 shows the result of repeatedly applying the operation of removing the dominant peak from the image. As can be seen, this process leads to the separation of the object from the background, enabling its further analysis using subsequent object recognition operations.

The presented chromatograms indicate that four chromatographic plates were used for the removal operation. As the number of plates increases, the generalization capability of the chromatographic algorithm decreases, which results in the need to perform the background removal process multiple times using the described procedure.

3.3. The operation of subtracting two chromatograms

The chromatogram subtraction operation presented below allows for highlighting the differences between two sets of image data, which is particularly useful in applications requiring change detection or background masking(Algorithms for Image Processing and Computer Vision Second Edition n.d.; Young et al. 1995).

The chromatogram subtraction operation can be used to extract differences between images. Algorithm 3 presents the successive steps of this procedure. It takes as input the image partitioning parameters and the number of chromatographic plates, which define the resolution, as well as a reference chromatogram to be subtracted from the chromatogram of image P. It is assumed that both chromatograms were generated using identical parameters of the chromatographic data separation algorithm.

Assumptions:
Let $P \in \mathbb{R}^{M \times N}$ be an image representing a mixture of substances.
Ch_{sub}: chromatogram to be subtracted.
Each sub-image has size $M_1 \times N_1$, no overlaps.
Let $K = \left\lfloor \frac{M}{M_1} \cdot \frac{N}{N_1} \right\rfloor$ be the number of substances.
Let L: number of theoretical plates (resolution of chromatograph).
Let $t_{\min}$ and $t_{\max}$: min and max retention time.
Let $\varepsilon \in [0, 1]$: allowed difference for peak subtraction.
Let $\Delta t = \frac{t_{\max} - t_{\min}}{L}$.
Input:
Image P, Chromatogram Ch_{sub}, L, $t_{\min}$, $t_{\max}$, ε
Output:
Chromatogram Ch_{out}, Modified image P'
Initialize empty multiset $T_P = \emptyset$; `// Retention times from image P`
foreach *non-overlapping sub-image* $s_i \subseteq P$ *of size* $M_1 \times N_1$ **do**
    Compute affinity to stationary phase: $a_i \leftarrow f(s_i)$;
    Compute retention time: $t_i \leftarrow \text{Round}\,(t_{\min} + a_i \cdot (t_{\max} - t_{\min}), \Delta t)$;
    Add t_i to T_P;
Group T_P into peaks: for each $t \in T_P$, let

$$Ch_P(t) = |\{x \in T_P : x = t\}|$$

Initialize $Ch_{out} = Ch_P$;
foreach $(t_s, h_s) \in Ch_{sub}$ **do**
    foreach $t_p \in dom(Ch_P)$ **do**
        if $|t_s - t_p| \leq \varepsilon$ **then**
            $Ch_{out}(t_p) \leftarrow \max(Ch_P(t_p) - h_s, 0)$;
            break;
Modify image P accordingly: remove peaks in Ch_P that matched Ch_{sub} within ε;
Let $P' \leftarrow P$ with substances corresponding to removed peaks deleted;
return Ch_{out}, P'

Algorithm 3. Algorithm for Subtracting Two Chromatograms.

The chromatogram subtraction algorithm performs a comparison and removal of common peaks between the chromatogram of the input image and the reference chromatogram (ChSub).

Initially, the image P of size $M \times N$ is divided into K non-overlapping subregions (of size $M_1 \times N_1$), which are treated as separate "substances." For each subregion, the retention time rt_i is calculated based on its affinity to the stationary phase, and then discretized to the nearest value $t_i = RT_i + k \cdot \Delta RT$, where $\Delta RT = (RT_{ax} - RT_i)/L$ and $k \in \{0, 1, \ldots, L\}$.

Next, the chromatogram Ch_P is constructed as a set of pairs (t, n), where n represents the number of subregions with retention time t. In a loop, each peak $y = (t, n)$ from the reference chromatogram ChSub = {(t, n)} is examined, and for each t, the corresponding m in Ch_P is reduced by n if $|t - t| \leq \varepsilon \cdot \Delta RT$. If after subtraction $m \leq 0$, the peak is removed from Ch_P. As a result, we obtain a chromatogram C without the common peaks.

Finally, the modified image P′ is reconstructed by removing from P all subregions whose retention times are associated with the removed peaks. The algorithm returns the updated chromatogram C and the image P′, which is ready for subsequent stages of analysis (Schmidt-Traub, Schulte, Seidel-Morgenstern 2020; Stilo et al., 2021).

The chromatogram subtraction operation can be applied in two key areas of image processing. The first is object extraction through background removal, particularly in cases where the complex structure of the background makes simple filtering of the most abundant subregions ineffective—especially when the extracted object occupies a larger portion of the image than the background itself. The second area is skeletonization, which involves representing the shape of an object as a thin contour. This serves as an important stage in the image recognition process.

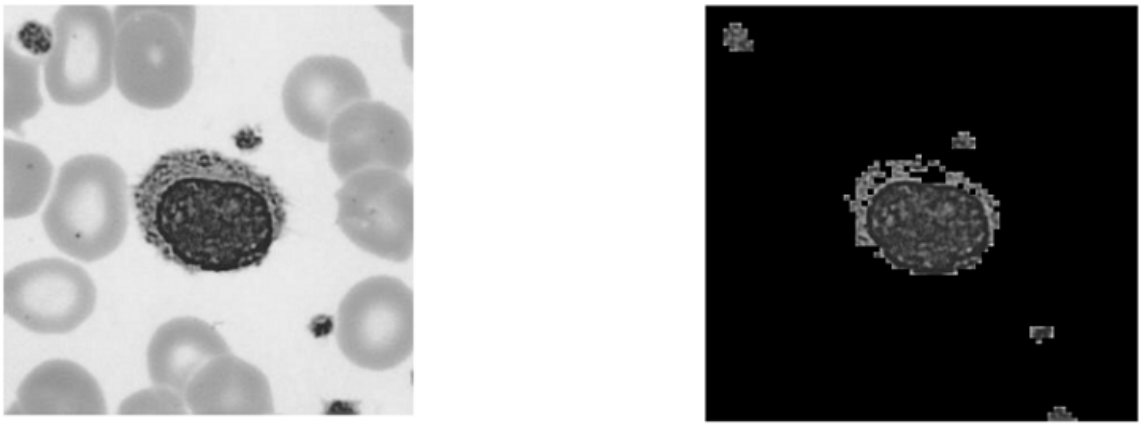

Figure 4. Background Removal via Chromatogram Subtraction.

Figure 4 presents the result of the chromatogram subtraction operation. The subtracted chromatogram was generated based on an image containing the background pattern. As shown on the right, the background has been removed, and the centrally positioned object is now prepared for further recognition processing.

Skeletonization of the object—its representation as thin contours—plays a key role in the image recognition process, as it enables shape-based identification. In the context of the chromatographic data separation method, achieving such an effect is possible through appropriate processing of the obtained chromatogram: selecting and analyzing peaks corresponding to the structural boundaries of the object, and subsequently reconstructing the contour based on the grouped image fragments.

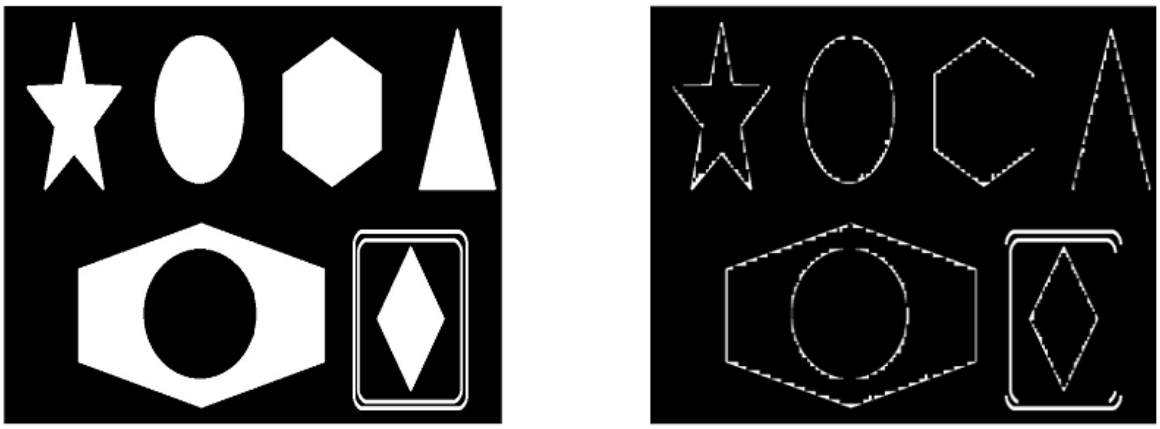

Figure 5. Skeletonization Resulting from Chromatogram Subtraction.

Figure 5 shows the result of the chromatogram subtraction operation when the reference chromatogram was generated from images containing filled objects intended for skeletonization. On the left side of the figure, the extracted contours of these objects are visible. However, it should be emphasized that the quality of the result is significantly influenced by the choice of parameters: the number of chromatographic plates in the data separation algorithm and the value of the ε parameter, which defines the allowable deviation between the retention times of compared peaks.

3.4. Algorithm for Removing Common Peaks Between Chromatograms

In image processing using techniques inspired by gas chromatography, one of the key operations can be the identification and removal of common features that appear in multiple images. These features—represented by peaks with similar retention times in chromatograms—may correspond to background elements, repetitive graphic patterns, or other information irrelevant to further analysis, such as classification or object recognition.

The common peak removal algorithm enables the extraction of unique components that are present only in selected images. It operates by comparing chromatograms calculated for a set of images using identical chromatographic parameters**.** Common peaks (i.e., those with retention times differing by less than a specified tolerance threshold) are identified and removed from the chromatograms of all images**,** except the one that contains the fewest peaks**,** which is treated as the reference point.

As a result, each modified image retains only unique features, which can significantly enhance the effectiveness of subsequent analysis stages—such as segmentation, pattern recognition, or machine learning.

Assume: Chromatograms $\mathcal{C} = \{Ch_1, \ldots, Ch_N\}$, each with same resolution, known ret. time span; $\varepsilon \in [0, 1]$.
Assume: Each Ch_i has peaks as tuples (t, f, s, p) where t: retention time, f: frequency, s: substance image, p: position.
Input : Chromatograms $\mathcal{C}$, parameter ε.
Output : Modified chromatograms $\mathcal{C}'$

if *all* Ch_i *have no peaks* **then**
  return $\mathcal{C}'$, $G = \emptyset$
$k \leftarrow \arg\min_i |Peaks(Ch_i)|$;
Let Ch_k be chromatogram with fewest peaks;
$\mathcal{C}' \leftarrow \mathcal{C}$;
foreach $i \in \{1, \ldots, N\}, i \neq k$ **do**
  foreach *peak* $p_i \in Peaks(Ch_i)$ **do**
    if $\exists p_k \in Peaks(Ch_k) : |t(p_i) - t(p_k)| \leq \varepsilon$ **then**
      remove p_i and its (f, s, p) from Ch'_i;
    end
  end
end
Let $P \leftarrow \bigcup_{i=1}^{N} Peaks(Ch'_i)$;
Initialize $G \leftarrow \emptyset$;
while $P \neq \emptyset$ **do**
  pick any $p \in P$;
  $C_p \leftarrow \{q \in P : |t(q) - t(p)| \leq \varepsilon\}$;
  $t_g \leftarrow \frac{1}{|C_p|} \sum_{q \in C_p} t(q)$;
  $h_g \leftarrow \sum_{q \in C_p} f(q)$;
  $G \leftarrow G \cup \{(t_g, h_g)\}$;
  $P \leftarrow P \setminus C_p$;
end
return $\mathcal{C}'$

Algorithm 4. Algorithm for Removing Common Peaks.

The algorithm is designed for processing multiple images that have undergone chromatographic separation, aiming to remove common features—i.e., peaks with similar retention times—from most chromatograms. This approach can be useful in extracting unique object features from images or in classification tasks where only differences between objects matter.

The algorithm input consists of a set of images P_1 , P_2 , ..., P_N, each of which has an associated chromatogram Ch_1 , Ch_2 , ..., Ch_N created based on dividing the image into subregions representing substances (i.e., image fragments). All chromatograms are generated using identical chromatographic resolution parameters—that is, the same number of shelves and the same minimum and maximum retention time values(Hage 1999; Martens et al., 2017; Robards, Ryan, 2021).

The first step of the algorithm is to identify the chromatogram with the smallest number of peaks, which is treated as the reference. Then, for each of the remaining chromatograms, its peaks are compared with the peaks of the reference chromatogram. If the difference between the retention times of the peaks does not exceed the given tolerance value epsilon (ε), these peaks are considered common and removed. In the final step, the chromatograms are recalculated by regrouping the substances based on their new retention times and contents.

The algorithm returns modified chromatograms along with their corresponding images, from which common information has been removed—enabling further processing focused solely on the unique features of each image.

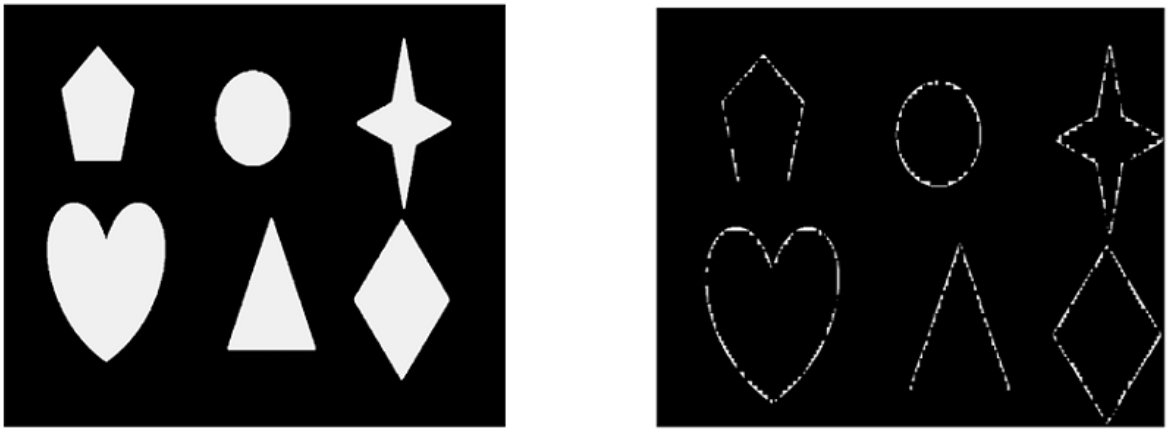

Figure 6. Input image and image resulting from the application of the common peak removal algorithm.

Figure 8 shows the result of the operation involving the removal of common peaks between chromatograms. As can be seen, this operation produced object contours, visible in the image on the left. Naturally, the quality of the obtained skeletons of individual objects is influenced by several parameters.

The parameters that significantly influence the quality of the obtained contours (skeletons) are primarily:

- The number of chromatographic shelves, which corresponds to the resolution of separation – the higher this number, the more precise the differentiation between substances (image fragments), but it can also lead to excessive data fragmentation.
- The retention time range – the minimum and maximum retention times define the limits within which peaks can be located; selecting these values appropriately helps to better separate important features.
- The value of the epsilon (ε) parameter – the tolerance for retention time differences when comparing peaks; too large a value may result in the removal of unique features, while too small a value may leave common information intact.
- The selection of these parameters should be tailored to the characteristics of the analyzed images and the processing goal (e.g., classification, segmentation, feature extraction).

3.5. Dilation algorithm

Dilation is one of the basic morphological operations used in image processing, mainly for binary images, although it can also be extended to grayscale images. It involves "expanding" objects by adding pixels to their boundaries. As a result, objects become larger, and small gaps, breaks, or irregularities in their structure are filled (Young et al., 1995).

The dilation algorithm processes a single input chromatogram, ChInput, which contains a set of peaks and associated sub-image fragments representing substances. For each peak i with a non-zero retention time, chromatographic separation of the corresponding sub-image fragment is performed using the function split_chromatogram, applying appropriate parameters such as sub-image dimensions, number of shelves, and minimum and maximum retention times.

Among the separation results, the sub-image fragment corresponding to the peak with the highest retention time and greatest frequency of occurrence is selected. Then, all other sub-image fragments in the chromatogram—including those related to peaks with zero retention time—are modified relative to the selected fragment (Giddings, 2017; Gupta, Biswas, 2023; Mondello et al., 2008).

Based on these modified sub-images, the chromatogram fragment for peak i is reconstructed. After processing all peaks, the algorithm rebuilds the entire input chromatogram using the reconstructed sub-image fragments and then chromatographically splits the chromatogram again based on the updated data.

As a result, the algorithm produces a chromatogram with dilated peaks, where objects (substances) are "expanded" or enlarged, which can facilitate further image processing and structural analysis.

Assumptions: Let Ch_{Input} be a chromatogram composed of peaks $P = \{p_1, ..., p_n\}$;
Each peak p_i contains:;
Retention time $t_i \in \mathbb{R}_{\geq 0}$;
Frequency $f_i \in \mathbb{N}$;
Set of subimages $S_i = \{s_1, ..., s_k\}$;
Set of positions $L_i = \{l_1, ..., l_k\}$;
Retention times are quantized as $\Delta t = \frac{t_{\max} - t_{\min}}{n_{slots}}$;
Each time t is rounded to the nearest multiple of Δt;
Function `split_chromatogram`(subimage, M, N, n_slots, t_{min}, t_{max}) returns a chromatogram;
Function `reconstruct_image`(chromatogram) reconstructs a subimage;
Input : Chromatogram Ch_{Input}, parameters $M_1, N_1, n_{slots}, t_{min}, t_{max}, M_2, N_2, n_{slots2}, t_{min2}, t_{max2}$
Output : Dilated chromatogram Ch_{Out}

foreach *peak* $p_i \in Ch_{Input}$ *with* $t_i > 0$ **do**
  foreach *subimage* $s \in S_i$ **do**
    $CH2 \leftarrow$ split_chromatogram(s, M_2, N_2, n_{slots2}, t_{min2}, t_{max2});
    Select peak $p^* \in CH2$ such that p^* has the maximum retention time and highest frequency;
    Let $s^{ref} \in p^*$ be the representative subimage;
    foreach *peak* $p \in CH2$ **do**
      foreach *subimage* $s' \in p$ **do**
        Modify s' using dilation based on s^{ref};
    $s^{new} \leftarrow$ reconstruct_image($CH2$);
    Replace s in S_i with s^{new};
$P_{global} \leftarrow$ reconstruct_image(Ch_{Input});
$Ch_{Out} \leftarrow$ split_chromatogram(P_{global}, M_1, N_1, n_{slots}, t_{min}, t_{max});

Algorithm 5. Gas chromatography-inspired dilation algorithm.

The algorithm presented in the pseudocode performs a dilation operation inspired by the principles of gas chromatography. For each peak in the input chromatogram with a non-zero retention time, all its sub-images are separated using the function split_chromatogram. Among the obtained peaks, the one with the highest retention time and the greatest frequency of occurrence is selected.

Next, the chosen sub-image is used to modify the remaining sub-images, including those belonging to peaks with zero retention time. The reconstructed sub-images are then reinserted back into the original peak.

After completing the operation for all peaks, the entire image is reconstructed and chromatographically re-divided. The final result is a reconstructed chromatogram after the dilation operation has been applied.

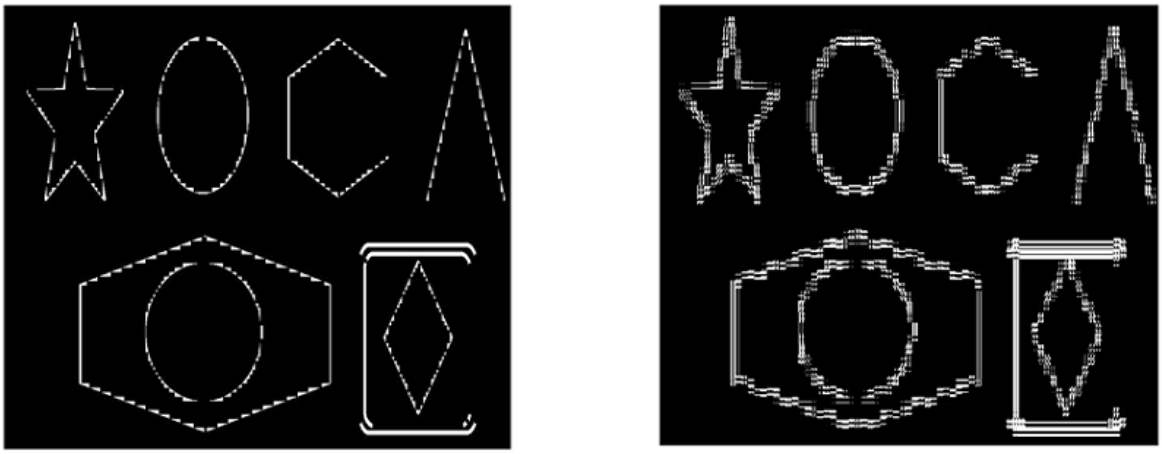

Figure 7. Input image and image resulting from applying the dilation algorithm.

Figure 7 shows the image before the dilation process on the left side. The image on the right has been processed using the presented dilation algorithm. As can be seen, the contours of individual objects have been thickened, and gaps in the object outlines have been filled. It is clear that the quality of the dilation process depends on the parameters described in the presented algorithm.

The dilation algorithm inspired by gas chromatography effectively expands and enhances the contours of objects in images, improving their clarity and coherence. Thanks to precise parameter selection and the use of chromatographic separation of sub-images, it is possible to eliminate gaps and defects in the structure of objects, which is important for subsequent stages of image analysis, such as recognition or classification. As a result, we obtain an image with clearer and more consistent outlines, which increases the efficiency of processing and interpreting visual data.

4. Classification

Classification of data is one of the fundamental problems in information analysis, involving the assignment of objects to predefined classes based on their features. In the context of processing data from chromatography, classification refers to the identification and assignment of individual retention peaks to specific patterns representing known chemical substances or groups of compounds.

The proposed classification algorithm is based on the paradigm of pattern matching and tolerant comparison of numerical features (in this case, retention times). The key assumption is that differences in retention times between actual data and patterns may result from inevitable instrumental errors, measurement noise, or slight changes in experimental conditions. Therefore, the comparison is not absolute but uses a relative tolerance threshold that allows

flexible matching (Blumberg, 2021; Chromatography: Definition, Working, and Importance in Various Industries n.d.; Gupta, Biswas 2023; Hage, 1999; Pierce et al., 2021).

The algorithm uses chromatograms represented as sets of numerical values describing peak positions (retention times) and searches these sets to identify the most probable matches to predefined patterns. This process is carried out using set operations and formal logical rules, enabling precise definition of assignment conditions.

Additionally, the algorithm takes into account the identification of peaks common to different pattern classes, which allows better characterization of ambiguous cases where a given peak may appear in more than one class. Such analysis enables later filtering of ambiguous matches or their inclusion in probabilistic classification.

The entire classification procedure is therefore based on three main paradigms:

- Relative comparison – tolerant matching of values considering relative error.
- Collective matching – classification performed based on a set of matches rather than individual values.
- Pattern structure – classes are represented by reference sets composed of multiple characteristic points (retention times), not just a single symbol.

This defined approach allows building a classifier resistant to noise and measurement uncertainties, while respecting the typical data structure of gas chromatography.

The presented classification algorithm is based on the idea of relative matching – a peak from the input chromatogram is considered a match to the pattern if its retention time falls within a specified tolerance threshold relative to the reference time. Furthermore, the algorithm distinguishes between unique and common peaks across pattern classes, enabling more precise multi-class analysis as well as identification of unmatched peaks, which may indicate the presence of unknown or contaminating chemical compounds.

This algorithm performs classification in five main stages: peak matching, detection of common pattern features, assignment of classes to peaks, aggregation of classification results, and identification of unmatched peaks. Such an approach allows for interpretable and flexible analysis of chromatographic data, which can be easily adapted to various levels of accuracy by adjusting the tolerance parameter.

assumptions: Let Ch_X be the input chromatogram with a finite set of retention times $T_X = \{t_1, t_2, ..., t_n\}$ and $W = \{W_1, W_2, ..., W_m\}$ be the set of reference chromatograms (templates), where W_k has its own retention time set T_k. A peak $t_i \in T_X$ matches $t_j \in T_k$ if $\frac{|t_i - t_j|}{t_j} \leq \varepsilon$, where ε is a matching threshold.
input : chromatogram Ch_X, set of reference chromatograms $W = \{W_1, W_2, ..., W_m\}$, tolerance ε
output : classification result, unmatched peaks, and peak-to-class assignments

```
foreach t_i ∈ T_X do
    initialize M_i = ∅;
    foreach W_k ∈ W do
        foreach t_j ∈ T_k do
            if |t_i − t_j| / t_j ≤ ε then
                add k to M_i;
                mark t_j in T_k as matched;
                break;
            end
        end
    end
end
foreach W_k ∈ W do
    initialize S_k = ∅, U_k = T_k;
    foreach t_j ∈ T_k do
        foreach W_l ∈ W, l ≠ k do
            if ∃t_m ∈ T_l such that |t_j − t_m| / t_m ≤ ε then
                add t_j to S_k (shared peaks);
                remove t_j from U_k (unique peaks);
            end
        end
    end
end
foreach t_i ∈ T_X do
    assign t_i to class set C_i = M_i;
end
foreach W_k ∈ W do
    let D_k = matched peaks in T_k;
    compute p_k = |D_k| / |T_k| · 100;
    store for class k: number of matched peaks, shared and unique peaks, and p_k;
end
foreach t_i ∈ T_X do
    if C_i = ∅ then
        add t_i to list of unmatched peaks;
    end
end
```

Algorithm 6. Classification algorithm based on gas chromatography principles.

The presented classification algorithm is based on comparing the retention times of peaks from the input chromatogram with the retention times recorded in the class patterns. This comparison relies on a relative tolerance of the difference, meaning that a given peak is assigned to a class if its retention time falls within a defined error threshold relative to the reference value. This approach allows accounting for natural deviations arising from experimental measurements.

The algorithm performs classification in five main steps: (1) matching peaks to patterns, (2) detecting common retention time features between classes, (3) assigning classes to peaks, (4) aggregating classification results, and (5) detecting unmatched peaks. Ultimately, each subimage representing a chemical compound in the image is either classified or marked as unidentified.

4.1. Initialization (lines 1-6)

From chromatogram ChX, a list of retention times (timesX, line 1) is extracted. Then, the number of input peaks nX and the number of reference classes nW are determined (lines 2-3). Matching structures, `matches` and `matched_in_patterns` (lines 4-6), are created to store the results of matches between peaks and patterns.

4.2. Peak matching (lines 7-18)

For each peak *i* in *ChX*, its retention time *tX* is retrieved (line 8). Then it is compared with each retention time *tW* in every class *k* (lines 9-11). The relative difference |tX - tW| / tW is calculated, and if it is less than or equal to ε, peak *i* is assigned to class *k* (lines 12-14). The match is recorded in both structures (lines 13-14), and the search within the current pattern ends (break, line 15).

4.3. Detection of common peaks between classes (lines 19-34)

For each pattern k, a structure wspolne_piki_info is created (lines 19-21). For each retention time t_1 in class k, its value is compared with retention times t_2 from other classes m (lines 22-28). If $|t_1 - t_2| / t_2 \leq \varepsilon$, these peaks are considered common (lines 26-27). Identified peaks and their sources are recorded in the wspolne_piki_info structure (lines 29-33).

4.4. Assignment of Classes to Peaks (lines 35-47)

A structure named peak_assignment is created (line 35). For each peak from ChX, the algorithm determines the set of classes it belongs to, based on matching with the retention times in the reference patterns (lines 36-44). The use of unique() (line 45) removes duplicate class assignments for each peak.

4.5. Aggregation of Results (lines 48-71)

For each class k, the number of matched peaks is calculated (number_matched, line 49), along with the match percentage (line 51). Then, the matched data is separated into two categories:

- Shared peak**s** (lines 52-56): these are peaks that match class k, but also appear in other classes within the tolerance ε – i.e., they are considered ambiguous.
- Unique peaks (lines 57-62): these are peaks that match only class k, and are not identified as shared with any other class.

Finally, both types of peaks along with summary statistics are stored in the classification structure (lines 63–71), which will be used for downstream analysis or reporting.

4.6. Detection of unmatched peaks (lines 72-78)

At the end, the algorithm identifies all peaks from ChX that have not been assigned to any class (lines 73-77). Their retention times and indices are stored in `unmatched_times` and `unmatched_indices`.

The algorithm classifies peaks (chemical substances) based on retention time, taking into account an error margin ε. It identifies both unique peaks, characteristic for only one class, and shared peaks common to multiple classes. Thanks to the assignment and classification structures, further comparative analysis of chromatograms is possible.

5. Classification Results of Images – Case Study

This chapter presents the results of classifier performance tests conducted on two independent datasets, differing in nature and potential practical application. The first case concerns the recognition of human leukocyte types based on synthetic images — a task directly related to medical applications such as automated blood diagnostics and laboratory support. The second case focuses on the analysis of geometric shapes, which is relevant in industrial settings, for example in component identification on production lines, sorting of elements, or quality control.

Both datasets were designed to enable precise evaluation of the classifier's ability to distinguish classes under both ideal conditions (training set) and noisy conditions (test set). The results presented below allow assessment of classification effectiveness in real-world contexts — both in medical and industrial environments.

5.1. Case 1 – Cell Identification

In the conducted study, a synthetic image dataset was used, representing five basic types of human leukocytes: basophils, eosinophils, neutrophils, monocytes, and lymphocytes. Although all these cells belong to the hematopoietic system and share similar general structures — including a nucleus, cytoplasm, and cell membrane — they differ significantly in shape, size, the number and arrangement of internal structures, as well as the texture of their interiors. These morphological differences are crucial for their classification process.

The dataset was entirely synthetically generated using a proprietary function designed to simulate realistic microscopic images of blood cells. The images are generated with random parameters responsible for varying shape, size, internal structure, and texture, allowing diverse samples for each class. This approach enabled the creation of a controlled yet sufficiently varied dataset, facilitating an effective evaluation of the classifier's performance.

The generated dataset was divided into two parts: a training set and a test set. The training set contains ten samples for each cell class, totaling fifty images. These samples were used to generate reference chromatograms—representative templates for each class—which serve as the basis for classifying new samples.

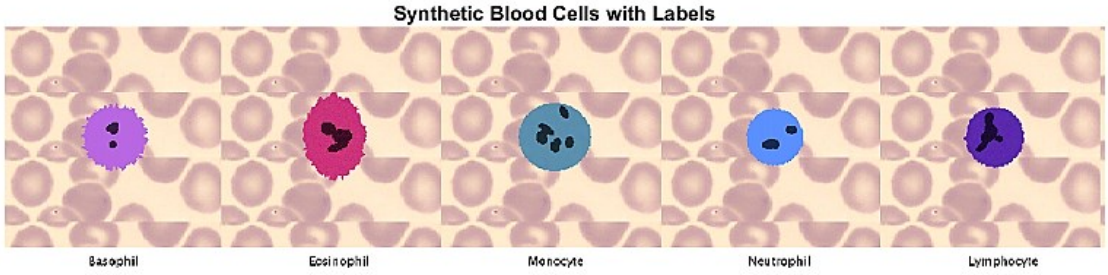

Figure 8. Synthetic Blood Cells with Labels.

The test set, which serves as the main element for evaluating the classifier's performance, contains thirty independent samples for each of the five classes, totaling one hundred fifty images. It was generated using a different set of random parameters, ensuring its independence from the training set and allowing for a realistic assessment of the classifier's generalization ability. Each image in the test set has a resolution of 360 × 360 pixels.

This design of the test set enables a controlled yet realistic evaluation of the classification system's performance and its robustness against morphological variability in blood cell images.

Table 1.
Classification results for the synthetic blood cell dataset

No.	Partition Type – Subimage Size	No. theoretical plate	(precision)
1	(30,30)	100 000	0,38
2	(20,20)	100 000	0,41
3	(10,10)	100 000	0,47
4	(5,5)	100 000	0,63
5	(4,4)	100 000	0,78
6	(3,3)	100 000	0,74
7	(4,4)	1000	0,52
8	(10,10)	1000	0,48
9	(20,20)	1000	0,64
10	(30,30)	1000	0,61
11	(4,4)	100	0,40
12	(10,10)	100	0,20
13	(20,20)	100	0,50
14	(30,30)	100	0,58

Source: own elaboration.

Based on the data presented in Table 1 regarding the classification of synthetic blood cells, a clear relationship can be observed between the size of the subimage, the theoretical number of bins (shelves), and the classification accuracy (precision). The classifier was designed based on an analogy to the operation of a gas chromatograph, where the number of bins determines the level of component separation. In the context of image processing, this translates to the degree of resolution of features extracted from divided image fragments.

The first six rows of the table (positions 1-6) show classification results for a very high number of bins (100,000), where the system should achieve the highest separation capability. Indeed, for large subimages (e.g., 30×30 pixels, row 1), the classification precision is low (0.38), suggesting that such large fragments are not sensitive enough to local changes in cell structure. Gradually reducing the subimage size leads to a noticeable increase in precision: (10×10) yields 0.47 (row 3), (5×5) reaches 0.63 (row 4), and (4×4) achieves 0.78 (row 5), which is the best result in the entire dataset. Further reduction to (3×3) (row 6) slightly worsens the result (0.74), possibly due to overfragmentation of features and increased sensitivity to noise.

Rows 7–10 illustrate the classifier's performance at a medium number of bins (1000). Here it is clearly visible that precision decreases compared to analogous subimage sizes at 100,000 bins. For example: (4×4), 1000 bins – 0.52 (row 7) vs. 0.78 at 100,000 bins (row 5); (10×10), 1000 bins – 0.48 (row 8) vs. 0.47 (row 3). Interestingly, for larger subdivisions (20×20 and 30×30), the results for 1,000 bins are higher (rows 9 and 10: 0.64 and 0.61, respectively) than at 100,000 bins (rows 2 and 1: 0.41 and 0.38). This may indicate that with fewer bins and larger fragments, the algorithm overfits less and demonstrates better generalization.

The last four rows (11-14) correspond to the lowest number of bins (100), which translates to the lowest feature extraction resolution. The results are mixed and generally lower. For example: (4×4), 100 bins – precision 0.40 (row 11), while (10×10) yields only 0.20 (row 12). Interestingly, (20×20) and (30×30) with precision 0.50 and 0.58 (rows 13 and 14) outperform the results obtained with the same sizes at 100,000 bins (rows 2 and 1). This might result from the fact that a low number of bins forces greater generalization, which, for large fragments, works better for some cell classes.

In summary, the most effective classification (row 5: 0.78) was achieved with an appropriately fine image subdivision (4×4) and a very high number of bins (100,000), allowing the capture of significant differences between cells of various sizes, shapes, and nuclear properties. It is also important to emphasize that before classification, each image was preprocessed by background removal using chromatogram subtraction and extraction of unique features (peaks). These steps enhanced contrast between structures and enabled the classifier to better distinguish diagnostically relevant feature.

5.2. Case 2 – Shape Analysis

In automation tasks, such as identifying parts on an assembly line, fast and reliable shape recognition based on images is crucial. Vision systems are used for this purpose, operating with pre-trained classifiers. To enable effective training and testing of these systems, a special dataset was prepared containing images of selected geometric shapes, representing different object classes.

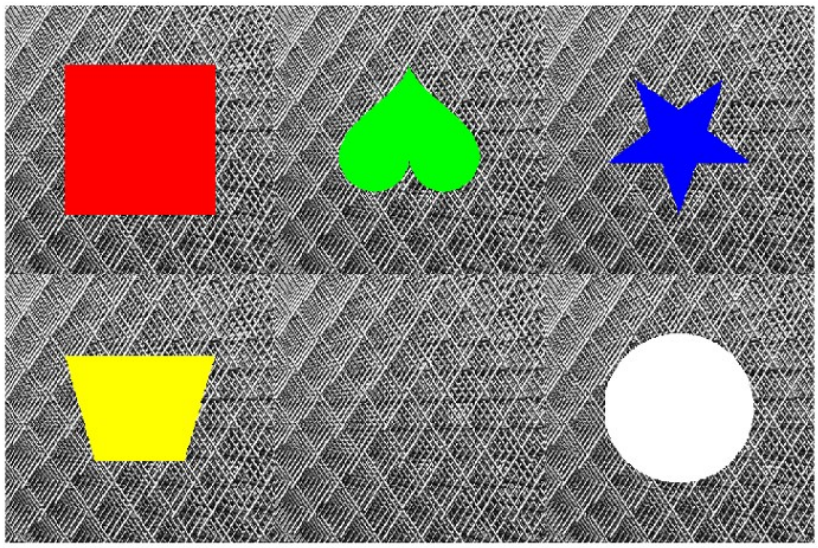

Figure 9. Dataset – geometric figures.

The dataset consists of two parts — training and testing. Both contain images of five classes of shapes: square, star, heart, circle, and trapezoid. The images are of uniform size and depict a single shape centered on a solid-colored background. The training set contains perfect representations of the shapes, without any distortions or noise, while the test set includes distorted versions — with random shape deviations and positional shifts. This setup allows testing under conditions similar to real industrial environments, where objects to be recognized may be shifted, rotated, or slightly deformed. Such a dataset structure enables evaluating the classifier's effectiveness not only in ideal conditions but also in scenarios requiring robustness to disturbances.

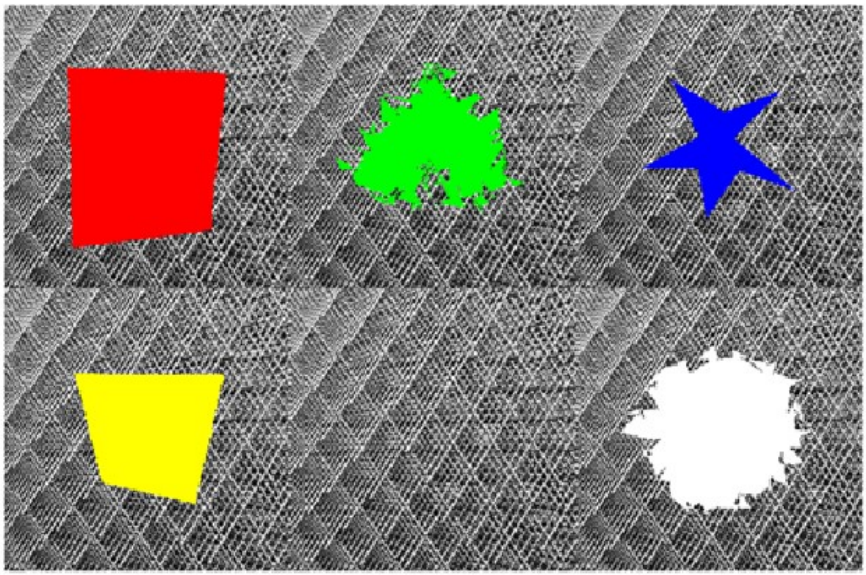

Figure 10. Dataset – distorted geometric figures.

The dataset used for classification consists of two parts: the training set and the test set. Both sets contain images depicting geometric shapes belonging to one of five classes: square, star, heart, circle, and trapezoid. Each shape is placed on a uniform black background. The images have a fixed resolution of 360×360 pixels, with the shapes always centered in the frame. The images are stored in RGB mode.

The training set contains 5 images, one for each class. All shapes in this set have perfect, regular forms — free from geometric distortions or noise. Their shape, proportions, and fill are clear and representative of their respective classes. These data serve as the basis for training the classifier, defining a clean and prototypical representation of each category.

The test set consists of 150 images, 30 for each of the five classes. Unlike the training set, shapes in the test set have undergone geometric deformations — their contours may be jagged, curved, or shifted. Such distortions aim to replicate realistic conditions where input data may deviate from the ideal prototype. This allows testing the classifier's robustness to irregularities and its ability to generalize.

Each shape in both sets has a similar size — about 200 pixels in the largest dimension. The colors assigned to the shapes are varied and consistent within each class, but they do not serve as the basis for classification — they only support visual distinction of examples.

The dataset is constructed to enable effective training and testing of the classifier under controlled conditions, but with an element of randomness and deformation that allows evaluation of the model's overall ability to recognize classes in less-than-ideal scenarios.

Table 2.
Classification results involving geometric figures

No.	Partition Type – Subimage Size	No. theoretical plate	(precision)
1	(60,60)	500	0,71
2	(40,40)	500	0,48
3	(20,20)	500	0,31
4	(10,10)	500	0,22
5	(60,60)	50	0,84
6	(40,40)	50	0,64
7	(20,20)	50	0,29
8	(10,10)	50	0,23
9	(60,60)	10	0,63
10	(40,40)	10	0,37
11	(20,20)	10	0,20
12	(10,10)	10	0,20

Source: own elaboration.

Analysis of the classification results for the second dataset highlights the significant impact of both the number of theoretical plates and the subimage size on the effectiveness of the classification algorithm based on an analogy to gas chromatography. In this dataset, an important feature is that the shapes representing different classes differ in their external form, while the interior texture remains constant within each class — shape variations mainly result from geometric distortions. Therefore, it is crucial that the classifier captures interior features rather than contours.

High number of plates – 500 (rows 1-4). A high number of plates (500) corresponds to a high separation capability of the algorithm, which means low generalization. This implies that the algorithm tries to distinguish subtle differences between samples as much as possible. In cases where variability arises only from shape, and the interior remains identical, such high separation can lead to overfitting and poor performance on distorted data.

Results confirm this:

- For the largest subimage size (60×60), precision reaches 0.71 (row 1), indicating that capturing the entire shape (both contour and interior) favors classification.
- However, as the subimage size decreases (down to 10×10), precision drops drastically (to 0.22 in row 4), showing that too small image fragments do not capture the shape's context well and cause errors.

Medium number of plates – 50 (rows 5-8) With 50 plates, the algorithm shows moderate generalization ability. Classification results are the best in the whole set — the highest precision (0.84) is achieved for the 60×60 partition (row 5). This can be explained by the fact that larger fragments allow capturing the whole shape and its contour, while the smaller number of plates avoids over-separation of minor distortions.

Precision decreases with decreasing subimage size:

- 40×40 — 0.64 (row 6),
- 20×20 — 0.29 (row 7),
- 10×10 — 0.23 (row 8).

This indicates that smaller subimages reduce the classifier's effectiveness in dealing with shape deformation differences, as the global object context is lost.

Low number of plates – 10 (rows 9-12). The lowest number of plates (10) represents very high generalization. The algorithm tries to capture only the most general fragment features. Under this setting, with large subimages (60×60), classification is still acceptable (0.63 in row 9) but then deteriorates sharply:

- 40×40 — 0.37 (row 10),
- 20×20 and 10×10 — only 0.20 (rows 11 and 12).

Such strong generalization means the algorithm cannot distinguish shapes properly, failing to separate classes that rely on contour deformation.

Final conclusions:

- The best result (precision 0.84) was obtained for the large subimage (60×60) and medium number of plates (50) — a compromise between capturing the whole shape and avoiding overfitting.
- Too small subimages cause loss of shape information, which is crucial here since the interior of the figure does not change.
- Too high a number of plates leads to overfitting on subtle differences from deformation, reducing effectiveness.
- Too low a number of plates (e.g., 10) results in poor discrimination between classes.

In this dataset, where class distinction is based on the shape of distorted figures with preserved interiors, the best results come from medium separation (50 plates) and full or nearly full coverage of the figure in a single subimage (60×60 or 40×40).

5.3. Summary of image classification results

This chapter presents the results of testing the effectiveness of an image classifier in two different applications: medical and industrial.

The first case involved recognizing types of human leukocytes based on synthetic microscopic images. Both the training and test data were generated in a controlled manner, enabling precise evaluation of the algorithm's performance. The highest classification precision (0.78) was achieved when the image was divided into very small fragments (4×4 pixels) and with a very high number of theoretical plates (100,000), which allowed capturing significant morphological differences between cells. Using too small fragments (e.g., 3×3) led to a decrease in precision, indicating increased susceptibility to noise. On the other hand, with lower numbers of plates, larger fragments yielded better results, favoring improved generalization, especially when the distinguishing features of the cells were more global in nature.

The second case focused on classifying simple geometric shapes in an industrial context, such as identifying components on an assembly line. The training set contained ideal shapes, while the test set included their distorted versions to simulate real-world conditions of vision systems. The best result (precision 0.84) was obtained with large image fragments (60×60

pixels) and a medium number of theoretical plates (50). This suggests that larger fragments better capture the overall shape of the object, and moderate separation capacity helps avoid overfitting. Too fine fragmentation and too low a number of plates significantly reduced effectiveness because the algorithm lost global context and struggled to correctly identify classes in the presence of distortions.

In summary, classification effectiveness depends both on the data structure and the algorithm parameters—particularly the fragment size and the number of theoretical plates. Good results are achieved when these parameters are matched to the data characteristics: in medical tasks, fine morphological details are crucial, while in industrial tasks, the overall object shape is more important.

6. Summary

The article presents an innovative approach to image analysis and classification based on the method of chromatographic data separation, inspired by classical gas chromatography principles. The authors treat an image as a mixture of compounds that can be separated according to their local properties, such as texture, color, and structure, assigning each image fragment a value corresponding to a retention time. As a result, a chromatogram is obtained—a histogram describing the distribution of these retention times—which serves as a compact representation of the image content. This representation is then used for classification by comparing it with reference chromatograms.

The described approach also includes a set of preprocessing operations that improve the quality of input data and extract significant features. Dominant components like the background are removed by subtracting chromatograms. Feature extraction involves eliminating common peaks present in many samples. Additionally, dilation operations help extract contours and enhance the presence of structures by simulated widening of components in the image.

In the experimental part, the classifier's effectiveness was tested in two independent scenarios. The first involved synthetic images of various human leukocyte types. Although these cells belong to the same biological system, they differ significantly in morphology, making classification particularly challenging. The second scenario concerned images of geometric shapes used in industrial contexts, such as part identification on production lines. Here, shape differences are key, while the texture inside the figures remains constant within each class.

Classification results show that the algorithm's performance strongly depends on parameter choices such as the number of theoretical plates, reflecting the system's resolving power, and the size of subregions into which the input image is divided. A high number of plates allows precise feature separation but may lead to overfitting when data variability is large. Conversely,

too few plates cause excessive generalization and loss of the ability to distinguish subtle differences. Experiments demonstrate that the best results occur with an appropriate balance between resolution and generalization capability, with optimal parameters varying depending on the dataset characteristics.

This chromatography-inspired approach can be an effective alternative to classical image recognition methods, especially in applications where selective extraction of unique features and robustness to input data disturbances are crucial.

References

1. *Algorithms for Image Processing and Computer Vision, Second Ed*.
2. Blumberg, L.M. (2021). Theory of Gas Chromatography. *Gas Chromatography*, pp. 19-97, doi: 10.1016/B978-0-12-820675-1.00026-5
3. *Chromatography: Definition, Working, and Importance in Various Industries*. Retrieved from: https://www.researchdive.com/blog/what-is-chromatography-how-does-it-work-and-where-is-it-used, 6.02.2024
4. *Digital image processing: principles and applications: Baxes, Gregory A: Free Download, Borrow, and Streaming: Internet Archive*. Retrieved from: https://archive.org/details/digitalimageproc0000baxe, 30.06.2025
5. Giddings, J.C. (2017). Dynamics of Chromatography: Principles and Theory. *Dynamics of Chromatography: Principles and Theory,* pp. 1-323. doi: 10.1201/9781315275871/DYNAMICS-CHROMATOGRAPHY-CALVIN-GIDDINGS
6. Gupta, M.K., Biswas, P.K. (2023). Chromatography: Basic Principle, Types, and Applications. *Basic Biotechniques for Bioprocess and Bioentrepreneurship*, pp. 173-182, doi: 10.1016/B978-0-12-816109-8.00010-6
7. Hage, D.S. (1999). Affinity Chromatography. *A Review of Clinical Applications*. Retrieved from: https://academic.oup.com/clinchem/article/45/5/593/5643177
8. Martens, J., Bhushan, R., Sajewicz, M., Kowalska, T. (2017). Chromatographic Enantioseparations in Achiral Environments: Myth or Truth? A Prevalent Concept of Chromatographic Enantioseparations and Its Later Modification. *Journal of Chromatographic Science*, *55(7)*, pp. 748-749, doi:10.1093/chromsci/bmx031
9. Mondello, L., Tranchida, P.Q., Dugo, P., Dugo, G. (2008). Comprehensive Two-Dimensional Gas Chromatography-Mass Spectrometry: A Review. *Mass Spectrometry Reviews*, *27(2)*, pp.101-124, doi: 10.1002/MAS.20158
10. Pierce, K.M., Trinklein, T.J., Nadeau, J.S., Synovec, R.E. (2021). Data Analysis Methods for Gas Chromatography. *Gas Chromatography*, pp. 525-546, doi: 10.1016/B978-0-12-820675-1.00007-1

11. Robards, K., Ryan, D. (2021). Principles and Practice of Modern Chromatographic Methods. *Principles and Practice of Modern Chromatographic Methods*, 1-518. doi: 10.1016/B978-0-12-822096-2.09993-X
12. Schmidt-Traub, H., Schulte, M., Seidel-Morgenstern, A. (2020). Preparative Chromatography: Third Edition. *Preparative Chromatography: 3rd Ed.,* 1-620, doi: 10.1002/9783527816347
13. Sparkman, O.D. (2005). Identification of Essential Oil Components by Gas Chromatography/Quadrupole Mass Spectroscopy Robert P. Adams. *Journal of the American Society for Mass Spectrometry*, *16(11)*. doi: 10.1016/j.jasms.2005.07.008
14. Stilo, F., Bicchi, C., Jimenez-Carvelo, A.M., Cuadros-Rodriguez, L., Reichenbach, S.E., Cordero, C. (2021). Chromatographic Fingerprinting by Comprehensive Two-Dimensional Chromatography: Fundamentals and Tools. *TrAC Trends in Analytical Chemistry*, *134:116133*, doi: 10.1016/J.TRAC.2020.116133
15. Święcicki, M. (2024). A Classification Algorithm Inspired by the Chromatographic Separation Mechanism Dedicated to the Classification of Variable-Length and Multi-Class Vectors. *International Journal on Informatics Visualization 8(1)*, pp. 368-77, doi: 10.62527/JOIV.8.1.2324
16. Święcicki, M. (2024). Application of chromatographic data separation methods in qualitative and quantitative data analysis. *Scientific Papers of Silesian University of Technology Organization and Management Series*, *213*, p. 569-590, doi: 10.29119/1641-3466.2024.213.40
17. Young, I.T., Gerbrands, J.J., Van Vliet, L.J. (1995). *Fundamentals of Image Processing.* Netherlands: Delft University of Technology.