UDC 004.93 **doi:** https://doi.org/10.20998/2522-9052.2025.2.02

Valerii Filatov, Anna Filatova, Anatolii Povoroznyuk

National Technical University “Kharkiv Polytechnic Institute”, Kharkiv, Ukraine

FUZZY IMAGE CLASSIFIER IN LARGE DYNAMIC DATABASES

Abstract. In today's rapidly growing visual information environment, the task of efficient image search and classification in large dynamic databases, which are updated daily with tens of thousands of new objects, is becoming especially relevant. Such databases are characterized not only by their significant size but also by a high degree of variability, which requires the development of algorithms capable of quickly and accurately recognizing distorted or modified versions of images in conditions of limited response time. **The subject of study** is a fuzzy classifier for clustering distorted versions of images in large dynamic databases. **The aim of this work** is to increase the accuracy of fast searches for distorted versions of images in large dynamic databases, in which the speed of adding information reaches 10-12 thousand images per day. **Methods used:** mathematical modeling, two-dimensional discrete cosine transform, image processing methods, decision-making methods, fuzzy mathematics. **The following results were obtained.** A fuzzy classifier for clustering distorted versions of images in large dynamic databases was developed. The experiments demonstrated that clustering distorted versions of images was sufficiently fast and cost-effective in terms of data volume and computational resource requirements. ROC analysis indicated the high quality of the developed fuzzy classifier.

Keywords: fuzzy image classifier; large dynamic database; distorted versions of images; computer system; fast search.

Introduction

Currently, the collection and processing of information, particularly in digital mass media systems, play a crucial role in shaping the information landscape of the world. A significant amount of information is contained in news feeds from electronic media, which include both textual and graphical components. Traditional approaches to processing news content predominantly rely on morphological and semantic analysis of text. However, images accompanying news are increasingly being recognized as an independent and significant source of information. This shift creates opportunities for clustering not only based on textual content but also on visual features. Furthermore, visual features are often more resilient to contextual variations than text keywords and are language-independent, allowing for the development of a single model for news items, rather than creating separate models for each language in which the news is published.

Thus, the problem of image clustering is highly relevant and has become a key component in the construction of news digest groups.

Literature analysis. Today, there are numerous commercial and research systems and services designed for image search and recognition, including the detection of distorted or modified versions [1–3]. Most existing solutions in the field of image retrieval focus on direct matching or global feature-based search [4, 5].

Commercial and cloud-based systems and services, such as Google Similar Images, TinEye, AntiDupl.NET, Microsoft Azure Computer Vision API, Clarifai, Amazon Rekognition, and others, demonstrate good performance in direct comparisons. However, despite their widespread use and functionality, these systems have several fundamental limitations that reduce their applicability to tasks involving large dynamic databases. These limitations include the inability to work with user databases, lack of support for streaming or bulk data entry, and low resistance to transformations such as geometric distortions, the use of decorative filters, partial cropping, logo overlay, or changes in format or resolution [6, 7]. Additionally, these systems often lack flexibility and transparency in their matching algorithms, as most operate as "black boxes" that do not allow users to customize image comparison parameters, a critical feature for scientific analysis and visual clustering tasks [8, 9].

Among the research and open-source systems, the following are noteworthy:

– ImageMatch (a Python library), based on SIFT/ORB descriptors and the Hamming distance, enables image comparison using local descriptors. Although the library can be integrated with its own database, it is not robust to strong distortions and noise.

– ImgSeek (a CBIR system in Python) searches for images with similar features using vector representations. It is no longer actively developed and relies on outdated algorithms.

– DeepDetect (a system in C++ and Python) employs convolutional neural networks for image processing, supporting classification, segmentation, and search. However, it requires complex configuration.

– Vise (Visual Search Engine) is based on LSH hashing and ORB descriptors and supports local database integration. A disadvantage is that it requires modification to fit user-specific tasks.

An analysis of the aforementioned systems and services allowed us to identify the main approaches to solving the problem of image search and recognition, particularly in conditions where images are subject to distortions, cropping, color changes, scaling, and other modifications.

Some of the earliest effective approaches were methods based on keypoint extraction using local descriptors, such as SIFT (Scale-Invariant Feature Transform), SURF (Speeded-Up Robust Features), BRIEF (Binary Robust Independent Elementary Features), ORB (Oriented FAST and Rotated BRIEF), BRISK (Binary Robust Invariant Scalable Keypoints), and FREAK (Fast Retina Keypoint) [10, 11]. These methods have demonstrated good performance, but they have limitations when working with large dynamic databases. A typical image (~640 × 480 in size) generates between 500 and 2000 keypoints. With approximately 1000 descriptors for each of 10000 images, around 10 million comparisons are required.

For example, extracting SIFT features from a single image can take 200-500 ms on a CPU and 10-50 ms on a GPU or when using optimized implementations (e.g., OpenCV with OpenCL), but it is still relatively slow. Additionally, the speed of comparison is influenced by the use of quadratic metrics. On average, comparing one image with a database of 1000 images takes several seconds on a CPU, especially without optimization. In large collections (>10000 images), the search speed drops sharply without optimization.

Algorithms based on binary descriptors, such as ORB, BRISK, or FREAK, are significantly faster due to storing less information (32 or 64 bytes per keypoint, compared to 512 bytes for SIFT) and using the Hamming distance for comparison, which is fast and efficient. However, the accuracy of these algorithms is considerably lower, especially when processing distorted images.

Another group of methods involves searching by hashed features [12, 13]. For example, commercial systems employ perceptual hashing techniques such as pHash, aHash, and dHash, which enable fast image comparison based on a 'visual fingerprint'. However, these methods are not robust to complex transformations, including rotations and scaling. As a result, these methods are primarily applied for rapid pre-filtering of potential matches.

The emergence of deep convolutional neural networks has significantly changed the field [14, 15]. Architectures such as ResNet, Inception, and VGG have become standard tools for extracting fixed-length feature vectors (feature embeddings) from images [16]. Nevertheless, deep models require substantial computational resources, and their effectiveness largely depends on the quality and representativeness of the training data.

Recent studies have demonstrated the potential of self-supervised learning and visual transformers (ViT). Models such as DINO, SimCLR, and CLIP (developed by OpenAI) show high robustness to distortions and are capable of aligning images with textual descriptions [17]. These models can reliably recognize images even under significant transformations but require powerful hardware infrastructure and are complex to deploy.

The analysis indicates that current systems and services for image clustering in large dynamic databases suffer from several limitations, including insufficient performance, limited accuracy, and proprietary implementations.

In [18], the authors proposed an image classifier that maintains performance regardless of the increasing volume of data in the database.

The general model of an image classifier can be represented as a tuple $IC = \langle I, C, S, R, O \rangle$, where $I = \{I_1, I_2, \dots, I_P\}$ is the set of images that need to be classified (collection of images); $C = \{C_1, C_2, \dots, C_K\}$ is a set of clusters (image classes), while $C_i \cap C_j = \varnothing$ $\forall i \neq j$; $S = \{\vec{S}_1, \vec{S}_2, \dots, \vec{S}_L\}$ is a set of image signatures; $R \subset C \times S$ is relationship between clusters and signatures; $O : I \to C$ is a clustering operation, which consists of transforming images, after which either an image $I_n \in I$ with a signature $\vec{S}_n \in S$ belongs to an existing cluster $C_k \in C$, or a conclusion is made about the need to create a new cluster $C_{K+1} \in C$ to which this image can be assigned, while one image can be assigned to only one cluster. The relation R has the following property: $\forall C_i \in C \exists \vec{S}_j \in S : (C_i, \vec{S}_j) \in R$.

Then a new image $I_n \in I$ with a signature $\vec{S}_n \in S$ belongs to an existing cluster $C_k \in C$ if there is such a signature $\vec{S}_l = (S_{l0}, \dots, S_{li}, \dots, S_{lM-1})$, the distance to which is minimal among all signatures and less than a given threshold ε_k. Otherwise, a new cluster C_{K+1} with the image $I_n \in I$ is created. Thus, the decision rule of the classifier has the following form

$$I_n \in \begin{cases} C_k, & \text{if } \exists (C_k, \vec{S}_l) : d(\vec{S}_n, \vec{S}_l) = \\ & = \min\limits_{p \neq n,\, p \in \overline{1,P}} d(\vec{S}_n, \vec{S}_p) < \varepsilon_k, 1 \le k \le K; \\ C_{K+1}, & \text{otherwise.} \end{cases} \quad (1)$$

It is proposed to use the metric of city blocks as a measure of the similarity of two vectors $\vec{a}$ and $\vec{b}$

$$d(\vec{a}, \vec{b}) = \sum_i |a_i - b_i| . \quad (2)$$

The computation of signatures in (1) based on the two-dimensional discrete cosine transform ensures the classifier's invariance to affine transformations of the original image. Nevertheless, research has shown that when classified images contain distortions such as changes in color gamut, slight shifts, or cropping by a few pixels, their signatures begin to deviate from the reference, leading to clustering errors when using the classifier proposed in [18]. These errors manifest either as the formation of additional clusters containing distorted images (Type I errors) or as the merging of dissimilar images into a single cluster (Type II errors). In such cases, the classifier's parameter settings do not allow for the simultaneous reduction of both Type I and Type II errors.

Therefore, the task of improving the classification quality for distorted versions of images in large databases remains highly relevant.

Problem statement and purpose of the study. The aim of this study is to enhance the accuracy of rapid search for distorted versions of images in large dynamic databases, where the data ingestion rate reaches 10–12 thousand images per day.

To achieve this goal, it is necessary to solve the following tasks:

– to develop a fuzzy classifier for clustering distorted versions of images in large dynamic databases;

– to conduct an experimental evaluation of the proposed fuzzy classifier.

Development of a fuzzy classifier

Studies on the classification quality using the proposed crisp classifier [1] have shown that the signature vector coordinates of distorted image versions may vary within a small range, leading to type I errors. At the same time, reducing the accuracy of determining the signature components $\vec{S}_l \in S$ through the use of quantization matrices $\mathbf{Q}_{gs}, \mathbf{Q}_r, \mathbf{Q}_g, \mathbf{Q}_b$ [18] or lowering the threshold ε_k in the decision rule (1), although decreasing type I errors, significantly increases type II errors (i.e., not only distorted versions of the image but also dissimilar images are clustered together).

This paper proposes the use of fuzzy mathematics to determine the degree of membership of an image to a given class.

Consider a class C_k consisting of visually similar images. Then, for the class C_k, we can define a reference image $I_{0k} \in C_k$ with a corresponding signature vector $\vec{S}_k = (S_{ki})_{i=0}^{M-1}$. Let there be a subclass C_k^{sub} of images that are potentially similar to the images of class C_k, and $C_k^{sub} \not\subset C_k$ according to the results of crisp clustering based on the decision rule (1).

To form a subclass, C_k^{sub} it is necessary to identify images $I_l \notin C_k$ whose signatures are similar to the signature $\vec{S}_k$ of the reference image $I_{0k} \in C_k$, i.e. each coordinate S_{li} of the signature $\vec{S}_l$ of the image I_l must belong to the interval $[S_{ki} - \delta_i; S_{ki} + \delta_i]$, where S_{ki} is the i-th coordinate of the signature $\vec{S}_k$ of the reference image I_{0k}; δ_{ki} is the permissible deviation for the i-th coordinate. To solve the given problem, we introduce the concept of a fuzzy signature vector $\tilde{\vec{S}}_k^{sub} = (\tilde{S}_{ki}^{sub})_{i=0}^{M-1}$ with a rectangular membership function

$$\mu_{\tilde{S}_{ki}^{sub}}(S_{li}) = \begin{cases} 1, & \text{if } S_{ki} - \delta_{ki} \le S_{li} \le S_{ki} + \delta_{ki}; \\ 0, & \text{otherwise,} \end{cases} \qquad (3)$$

where δ_{ki} is the width parameter of the fuzzy set $\tilde{S}_{ki}^{sub}$.

The paper then proposes the following decision rule for forming the subclass C_k^{sub}

$$I_l \in \begin{cases} C_k^{sub}, & \text{if } \mu_{\tilde{S}_{ki}^{sub}}(S_{li}) = 1 \ \forall i; \\ C_L, & \text{otherwise,} \end{cases} \qquad (4)$$

where $C_L \not\subset C_k$ is the class to which the image was assigned I_l at the stage of crisp classification using the decision rule (1).

Let us describe the reference image I_{0k} by a fuzzy normalized signature vector $\tilde{\vec{S}}'_k = (\tilde{S}'_{ki})_{i=0}^{M-1}$ with a Gaussian membership function

$$\mu_{\tilde{S}_{ki}}(S'_{ni}) = \exp\left(-\frac{(S'_{ni} - S'_{ki})^2}{2\sigma_{ki}^2} \right), \qquad (5)$$

where $S'_{ni} \in [0;1]$ is the normalized value of the i-th coordinate of the crisp signature vector $\vec{S}_n = (S_{ni})_{i=0}^{M-1}$ of the image $I_n \in C_k^{sub}$ to be classified; C_k^{sub} is the subclass of images potentially similar to the reference image $I_{0k} \in C_k$; $S'_{ki} \in [0;1]$ is the center of the fuzzy set, which is determined by the value of the i-th coordinate of the crisp vector of the signature $\vec{S}_k = (S_{ki})_{i=0}^{M-1}$ of the reference image I_{0k}; σ_{ki} is the width parameter of the fuzzy set $\tilde{S}_{ki}^{sub}$.

Normalization of the coordinate values of the signature vector can be performed as follows [19]:

$$S'_i = \frac{S_i - \min_i(S_i)}{\max_i(S_i) - \min_i(S_i)}. \qquad (6)$$

Then, the problem of fuzzy classification of the image I_n is reduced to assessing the degree of membership of a crisp normalized signature vector $\vec{S}'_n$ to a fuzzy normalized signature vector $\tilde{\vec{S}}'_k$.

There are various methods for constructing a fuzzy decision rule to assess the membership of a crisp vector to a fuzzy vector [20, 201]. Let there be a crisp vector $\vec{S} = (S_i)_{i=0}^{M-1}$, and a fuzzy vector $\tilde{\vec{S}} = (\tilde{S}_i)_{i=0}^{M-1}$, where each coordinate of the fuzzy vector represents a fuzzy set with a membership function $\mu_{\tilde{S}_i}(S_i) \in [0;1]$.

The classical approach is to calculate the minimum value coordinate-wise $\mu(\vec{S} \in \tilde{\vec{S}}) = \min_i \mu_{\tilde{S}_i}(S_i)$. This approach reflects the weakest degree of membership; that is, if at least one coordinate has a low membership value, the entire vector is considered to have a low degree of membership. This method of defining the intersection operation is applicable in problems where strict control over all features is important.

Given that the method for determining the image signature is based on the two-dimensional discrete cosine transform [18], a strong deviation in a single coordinate should not be decisive in the intersection operation. Therefore, this method is not suitable for solving the problem.

An alternative approach involves computing the average degree of membership across all components of the vector:

$$\mu(\vec{S} \in \tilde{\vec{S}}) = \frac{1}{M} \sum_{i=0}^{M-1} \mu_{\tilde{S}_i}(S_i). \qquad (7)$$

In this case, an error in one coordinate can be compensated by the others, meaning there is no single critical coordinate. This method is less strict than the minimum-based approach, as it considers all components, even if one of them deviates significantly from the reference.

Then, the final solution is as follows:

$$I_l \in \begin{cases} C_k, & \text{if } \mu(\vec{S}'_l \in \tilde{\vec{S}}'_k) \ge \tau; \\ C_L, & \text{otherwise,} \end{cases} \qquad (8)$$

where $\tau \in (0;1)$ is the similarity threshold.

As a result, either the image I_l is added to class C_k, or it remains in the original class C_L, to which it was assigned during the crisp classification stage using the decision rule (1).

Software implementation of a fuzzy classifier

For fast and precise search in large databases, it was proposed in [18] to use a signature of up to 10 elements.

In this paper, we propose reducing the signature length to 6 elements during the precise search stage to increase speed.

Meanwhile, during the fuzzy search stage, the signature is extended to 20 elements.

The software implementation of the decision rule (4), taking into account the membership function (3), in PHP using the MySQL DBMS is as follows:

```
$deviation = 15
$sql = "SELECT COUNT(*) AS cnt, vr0, vr1, vr2, vr3, vr4, vr5,
avg(vr6) vr6, avg(vr7) vr7, avg(vr8) vr8, avg(vr9) vr9, avg(vr10) vr10, avg(vr11) vr11,
avg(vr12) vr12, avg(vr13) vr13,avg(vr14) vr14, avg(vr15) vr15,avg(vr16) vr16, avg(vr17) vr17,
avg(vr18) vr18, avg(vr19) vr19
FROM dct_test
GROUP BY vr0, vr1, vr2, vr3, vr4, vr5 ORDER BY cnt DESC";
$query = dbl_query($sql);
while ($res = dbl_fetch_assoc($query)) {
    $vector_target = $res;
    unset($vector_target['cnt']);
    $conditions = [];
    $key = [];
    foreach (['vr0', 'vr1', 'vr2', 'vr3', 'vr4', 'vr5'] as $key) {
        $fields[]=$key." != '".$res[$key]."'";
        $conditions[] = "(" .
            "({$key} BETWEEN " . ($res[$key] - $deviation) . " and " . ($res[$key] + $deviation) . "))";
    }
    $sql_similar = "SELECT vr0, vr1, vr2, vr3, vr4, vr5, vr6, vr7, vr8, vr9, vr10, vr11,
    vr12, vr13, vr14, vr15, vr16, vr17, vr18, vr19
    FROM dct_test WHERE " . implode(" AND ", $conditions) . " and ". implode(" AND ", $fields);
        $query_similar = dbl_query($sql_similar);
    while ($vector_test = dbl_fetch_assoc($query_similar)) {
        $m = test_m($vector_target, $vector_test);
    }
}
```

In the proposed implementation, signatures of the images $I_{0k} \in C_k$ are determined as the average value of the coordinates of the image signatures that belong to a given class. The user functions dbl_fetch_assoc() and dbl_query() are used to implement prepared queries to the database (which are not discussed in this article). The specificity of the rectangular membership function (3) implementation is that the database selection with BETWEEN conditions, combined using AND, takes up to 100 ms for indexed columns and up to 500 ms for non-indexed columns, which is quite acceptable for solving the problem. To implement the calculation of the average value of the membership degrees according to (7), the following function can be used:

```
function fuzzy_rule ($vector1, $vector2, $b)
{
    $mu = [];
    $vector1 = norm($vector1);
    $vector2 = norm($vector2);
    foreach ($vector1 as $i => $x) {
        $diff = ($x - $vector2[$i]) / $b;
        $mu[] = exp(- ($diff ** 2) / 2);
    }
   return  array_sum($mu1) / count($vector1);
}
```

The proposed implementation uses a user-defined function norm(), which performs normalization according to expression (6).

Experimental verification of fuzzy classifier

In this work, an experimental test of the developed fuzzy classifier was conducted on a dataset with the following characteristics: number of images – 804724; all images are colored; number of clusters – 445999.

The full characteristics of the dataset are provided in [18].

At the stage of crisp clustering, in decision rule (1), it was assumed that $\varepsilon_k = 0$, meaning that completely identical images or images that cannot be visually distinguished are grouped into the same clusters.

To form the subclasses C_k^{sub}, the width parameter of the fuzzy set $\tilde{S}_{ki}^{sub}$ in (3) was chosen as $\delta_i = 15 \forall i$. For calculating the coordinates of the vector $\tilde{\tilde{S}}'_k$, the width parameter in (5) was chosen as $\sigma_{ki} = 0.01 \forall i$.

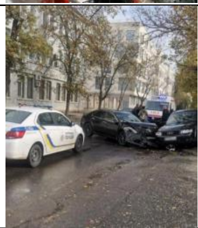
A comparative analysis was conducted between the crisp classifier developed in [18] and the fuzzy classifier proposed in this work, for images described by signature vectors of length $M = 20$. Table 1 presents the results of some experiments.

Experiment No.1 shows images with different levels of cropping (1.1–1.3), marked as 'add/crop', as well as an example of a dissimilar image (1.4), marked as 'not similar', from the subclass of potentially similar images. In addition, image 1.3 is flipped horizontally, marked as 'flip'.

Experiment No. 2 shows a dissimilar image (marked as 'not similar') from the class of potentially similar images with a high degree of membership.

Experiment No. 3 presents images that differ from each other both in cropping and in some additional details (marked as 'crop&change').

Table 1 – **Experimental results**

No.	Image I_{0k}	$\vec{S}_k$	Image I_l	$\vec{S}_l$	$d(\vec{S}_k, \vec{S}_l)$	$\mu(\vec{S}_l' \in \tilde{\vec{S}}_k')$
1.1 (add)		(636, 762, 771, 459, 925, 431, 94, 526, 553, 130, 72, 160, 323, 121,98, 101, 83, 104, 90, 109)		(634, 759, 767, 454, 920, 429, 94, 523, 547, 123, 82, 152, 319, 121, 95, 115, 94, 100, 89, 106)	95	0.916
1.2 (crop)				(647, 772, 783, 462, 936, 431, 84, 525, 555, 126, 78, 154, 318, 108, 109, 103, 89, 98, 79, 124)	145	0.911
1.3 (flip, crop)				(648, 774, 784, 463, 938, 432, 85, 526, 556, 126, 78, 154, 319, 109, 109, 103, 90, 98, 80, 124)	150	0.919
1.4 (not similar)				(634, 762, 767, 451, 921, 438, 110, 530, 542, 123, 70, 145, 312, 133, 82, 85, 91, 77, 77, 96)	196	0.647
2.1 (not similar)		(787, 933, 928, 487, 1099, 506, 86, 592, 572, 42, 190, 42, 295, 92, 122, 166, 234, 8, 22, 156)		(796, 943, 937, 485, 1109, 508, 82, 595, 572, 35, 187, 37, 300, 90, 122, 137, 228, 5, 29, 153)	129	0.855
3.1 (crop& change)		(764, 877, 885, 423, 1017, 399, 63, 463, 486, 29, 240, 35, 220, 73, 267, 176, 279, 20, 36, 309)		(769, 886, 894, 433, 1029, 408, 56, 472, 496, 21, 238, 29, 222, 68, 264, 180, 280, 26, 42, 308)	124	0.957

Research has shown that for a crisp classifier with decision rule (1), increasing the signature length does not simultaneously reduce both type 1 and type 2 errors. If the threshold ε_k is increased, type 1 errors decrease, while type 2 errors increase.

In addition, it is impossible to select a single threshold ε_k for all classes, which significantly complicates the implementation of the algorithm due to the need to optimize a very large number of parameters. Thus, the distances calculated by (2) between similar images in experiment No. 1 (in Table 1 $d(\vec{S}_1, \vec{S}_{1.2}) = 145$ for image 1.2 and $d(\vec{S}_1, \vec{S}_{1.3}) = 150$ for image 1.3) are greater than the distance between dissimilar images in experiment No. 2 (in Table 1 $d(\vec{S}_2, \vec{S}_{2.1}) = 129$ for image 2.1).

It is obvious that the more similar the images are to each other, the greater the membership degree (7) and the smaller the distance (2). However, during the analysis of the experimental data, it was noted that a smaller value of the metric (2) (column $\mu(\vec{S}_l' \in \tilde{\vec{S}}_k')$ in Table 1) does not always correspond to a larger value of the membership function (7) (column $d(\vec{S}_k, \vec{S}_l)$ in Table 1).

For example, in experiment No. 1 $\mu(\vec{S}_{1.2}' \in \tilde{\vec{S}}_1') = 0.911 < \mu(\vec{S}_{1.3}' \in \tilde{\vec{S}}_1') = 0.919$, and at the same time $d(\vec{S}_1, \vec{S}_{1.2}) = 145 < d(\vec{S}_1, \vec{S}_{1.3}) = 150$.

This discrepancy becomes even more pronounced when comparing the calculation results for images from different classes.

Thus, $\mu(\vec{S}'_{1.1} \in \tilde{\tilde{\vec{S}}}'_1) = 0.916 < \mu(\vec{S}'_{3.1} \in \tilde{\tilde{\vec{S}}}'_3) = 0.957$, although $d(\vec{S}_1, \vec{S}_{1.1}) = 95 < d(\vec{S}_3, \vec{S}_{3.1}) = 150$ (Table 1).

The problem of being unable to select a single threshold in the decision rule (1) is due to the non-normalized values of the metric (2).

In contrast, choosing the threshold τ in decision rule (8) is a simpler task, as the values of the membership function (7) are normalized and have a clear physical interpretation.

The conducted ROC analysis showed the high quality of the developed fuzzy classifier (Fig. 1).

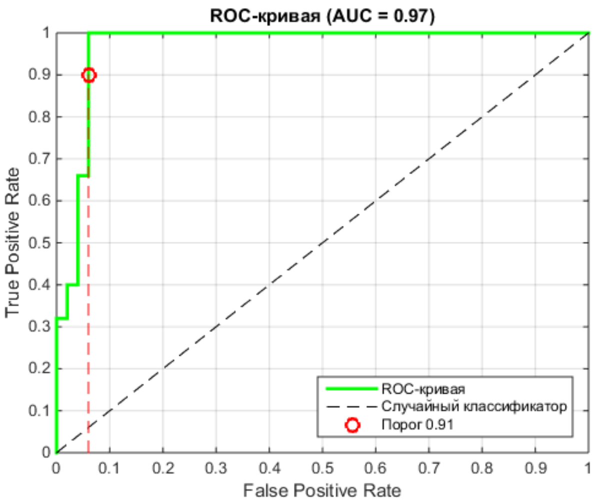


Fig. 1. ROC curve of the fuzzy classifier (8)

From the analysis of experimental results, the optimal threshold was determined to be $\tau \in [0.9, 0.91]$.

Conclusions

In this work, a fuzzy classifier has been developed to effectively cluster distorted versions of images in large dynamic databases. To ensure high processing speed, a two-stage classification approach is proposed. At the first stage, a fast crisp classifier is employed to detect identical or visually indistinguishable images. At the second stage, the developed fuzzy classifier is applied to identify distorted variants of images that could not be reliably clustered using the crisp method.

A complete software implementation of the fuzzy classifier has been carried out. Special attention was given to the optimization of membership function implementations, enabling efficient SQL queries in large databases.

The use of rectangular and Gaussian membership functions was evaluated with regard to execution time and accuracy.

Practical testing has shown that the proposed approach enables fast and resource-efficient clustering, even in databases containing hundreds of thousands of images.

Experimental results, including ROC analysis (AUC = 0.97), demonstrated the high classification quality of the fuzzy approach, significantly outperforming the crisp classifier in scenarios involving image distortions such as cropping, flipping, and content variation.

Future research will focus on fine-tuning the parameters of the fuzzy classifier, particularly the membership function width and threshold values.

Another promising direction is the adaptation of the proposed methodology for classifying other types of structured data, including the automatic classification and grouping of news content in digital media platforms.

REFERENCES

1. Khan, O. R. and Bhat, J. I. (2023), “Delving into the Depths of Image Retrieval Systems in the Light of Deep Learning: A Review”, *Indian Journal of Science and Technology*, vol. 16 (34), pp. 2693–2702, doi: https://doi.org/10.17485/IJST/v16i33.1341
2. Chung, H., Lee, N., Lee, H., Cho, Y. and Woo, J. (2023), “GuaRD: Guaranteed robustness of image retrieval system under data distortion turbulence”, *PLoS One*, vol. 18(9): e0288432, doi: http://dx.doi.org/10.1371/journal.pone.0288432
3. Garg, Shruti (2018), “A Study of Retrieval Methods of Multi-Dimensional Images in Different Domains”, *International Journal of Advanced Computer Science and Applications*, vol. 9 (10), pp. 342–352, doi: http://dx.doi.org/10.14569/IJACSA.2018.091041
4. Varish, N. (2022), “A modified similarity measurement for image retrieval scheme using fusion of color, texture and shape moments”, *Multimed Tools*, vol. 81, pp. 20373–20405, doi: https://doi.org/10.1007/s11042-022-12289-1
5. Shihao, Shao, Kaifeng, Chen, Arjun, Karpur, Qinghua, Cui, Andre, Araujo and Bingyi, Cao (2023), “Global Features are All You Need for Image Retrieval and Reranking”, *Computer Vision and Pattern Recognition*, pp. 1–13, doi: https://doi.org/10.48550/arXiv.2308.06954
6. Datta, R., Joshi, D., Li, J., and Wang, J. Z. (2008), “Image retrieval: Ideas, influences, and trends of the new age”, *ACM Computing Surveys (CSUR)*, vol. 40 (2, article 5), pp. 1–60, doi: https://doi.org/10.1145/1348246.1348248
7. Zhang, D. and Lu, G. (2004), “Review of shape representation and description techniques”, *Pattern Recognition*, vol. 37 (1), pp. 1–19, doi: https://doi.org/10.1016/j.patcog.2003.07.008
8. Kamakshi, V. and Krishnan, N. C. (2023), “Explainable Image Classification: The Journey So Far and the Road Ahead”, *AI*, vol. 4 (3), pp. 620–651, doi: https://doi.org/10.3390/ai4030033
9. Rudin, C. (2019), “Stop Explaining Black Box Machine Learning Models for High Stakes Decisions and Use Interpretable Models Instead”, *Nature Machine Intelligence*, vol. 1 (5), pp. 206–215, doi: http://dx.doi.org/10.1038/s42256-019-0048-x
10. Isik, M. (2024), “Comprehensive empirical evaluation of feature extractors in computer vision”, *PeerJ Computer Science*, 10:e2415, doi: https://doi.org/10.7717/peerj-cs.2415
11. Wu, S., Oerlemans, A., Bakker, E.M. and Lew, M.S. (2017), “A comprehensive evaluation of local detectors and descriptors”, *Signal Processing: Image Communication*, vol. 59, pp. 150–167, doi: https://doi.org/10.1016/j.image.2017.06.010

12. Yilmaz, A. (2025), "Bilateral Cross Hashing Image Retrieval Based on Principal Component Analysis", *Research Article-Computer Engineering and Computer Science*, pp. 1–18, doi: https://doi.org/10.1007/s13369-025-10135-8
13. Biswas, R. and Blanco-Medina, P. (2021), "State of the Art: Image Hashing", *Computer Vision and Pattern Recognition*, pp. 1–8, doi: https://doi.org/10.48550/arXiv.2108.11794
14. Singh, D., Mathew, J., Agarwal, M. and Govind M. (2023), "DLIRIR: Deep learning based improved Reverse Image Retrieval", *Engineering Applications of Artificial Intelligence*, vol. 126 (A), article number 106833, doi: https://doi.org/10.1016/j.engappai.2023.106833
15. Kubany, A., Ben, Sh., Ruben, I., Ohayon, S., Shmilovici, A., Rokach, L. and Doitshman, T. (2020), "Comparison of state-of-the-art deep learning APIs for image multi-label classification using semantic metrics", *Expert Systems with Applications*, vol. 161, 113656, doi: https://doi.org/10.1016/j.eswa.2020.113656
16. Shafiq, M. and Gu, Z. (2022), "Deep Residual Learning for Image Recognition: A Survey", *Applied Sciences*, vol. 12(18), 8972, doi: https://doi.org/10.3390/app12188972
17. Zhu, Y., Xu, H., Du, A. and Wang, B. (2024), "Image–Text Matching Model Based on CLIP Bimodal Encoding", *Applied Sciences*, vol. 14 (22), 10384, doi: https://doi.org/10.3390/app142210384
18. Filatov, V., Filatova, A., Povoroznyuk, A. and Omarov, S. (2024), "Image classifier for fast search in large databases", *Advanced Information Systems*, vol. 8, no. 2, pp. 12–19, doi: https://doi.org/10.20998/2522-9052.2024.2.02
19. Dun B., Zakovorotnyi, O. and Kuchuk, N. (2023), "Generating currency exchange rate data based on Quant-Gan model", *Advanced Information Systems*, vol. 7, no. 2, pp. 68–74, doi: http://dx.doi.org/10.20998/2522-9052.2023.2.10
20. Kamthan, Sh., Singh, H. and Meitzler, T. (2022), "Hierarchical fuzzy deep learning for image classification", *Memories - Materials, Devices, Circuits and Systems*, vol. 2, 100016, doi: https://doi.org/10.1016/j.memori.2022.100016
21. Li, Z., Li, L., Yan, K. and Zhang, C. (2017), "Automatic image annotation using fuzzy association rules and decision tree", *Multimedia Systems*, vol. 23, pp. 679–690, doi: https://doi.org/10.1007/s00530-016-0530-9

Received (Надійшла) 20.01.2025
Accepted for publication (Прийнята до друку) 02.04.2025

ABOUT THE AUTHORS / ВІДОМОСТІ ПРО АВТОРІВ

Філатов Валерій Володимирович – аспірант кафедри комп'ютерної інженерії та програмування, Національний технічний університет "Харківський політехнічний інститут", Харків, Україна;
Valerii Filatov – PhD Student of Computer Engineering and Programming Department, National Technical University "Kharkiv Polytechnic Institute", Kharkiv, Ukraine;
e-mail: filatov@mail.com; ORCID Author ID: https://orcid.org/0009-0007-7762-1517;
Scopus ID: https://www.scopus.com/authid/detail.uri?authorId=59178850500.

Філатова Ганна Євгенівна – доктор технічних наук, професор, професорка кафедри комп'ютерної інженерії та програмування, Національний технічний університет "Харківський політехнічний інститут", Харків, Україна;
Anna Filatova – Doctor of Technical Sciences, Professor, Professor of Computer Engineering and Programming Department, National Technical University "Kharkiv Polytechnic Institute", Kharkiv, Ukraine;
e-mail: filatova@gmail.com; ORCID Author ID: https://orcid.org/0000-0003-1982-2322;
Scopus ID: https://www.scopus.com/authid/detail.uri?authorId=56448583600.

Поворознюк Анатолій Іванович – доктор технічних наук, професор, професор кафедри комп'ютерної інженерії та програмування, Національний технічний університет "Харківський політехнічний інститут", Харків, Україна;
Anatolii Povoroznyuk – Doctor of Technical Sciences, Professor, Professor of Computer Engineering and Programming Department, National Technical University "Kharkiv Polytechnic Institute", Kharkiv, Ukraine;
e-mail: ai.povoroznjuk@gmail.com; ORCID Author ID: https://orcid.org/0000-0003-2499-2350;
Scopus ID: https://www.scopus.com/authid/detail.uri?authorId=55225664000.

Нечіткий класифікатор зображення у великих динамічних базах даних

В. В. Філатов, Г. Є. Філатова, А. І. Поворознюк

Анотація. В сучасних умовах стрімкого зростання обсягів візуальної інформації особливої актуальності набуває задача ефективного пошуку та класифікації зображень у великих динамічних базах даних, які щодня поповнюються десятками тисяч нових об'єктів. Такі бази характеризуються не лише значними розмірами, але й високим ступенем варіативності, що вимагає розробки алгоритмів, здатних до швидкого та точного розпізнавання спотворених або модифікованих версій зображень в умовах обмеженого часу відгуку. **Предмет дослідження:** нечіткий класифікатор для кластеризації спотворених версій зображень у великих динамічних базах даних. **Метою дослідження** є підвищення точності швидкого пошуку спотворених версій зображень у великих динамічних базах даних, у яких швидкість додавання інформації сягає 10–12 тисяч зображень на добу. **Методи, що використовуються:** математичне моделювання, двовимірне дискретне косинусне перетворення, методи обробки зображень, методи прийняття рішень, нечітка математика. **Отримані результати.** Розроблено нечіткий класифікатор для кластеризації спотворених версій зображень у великих динамічних базах даних. Проведені експерименти показали, що кластеризація спотворених версій зображень виявилася досить швидкою та маловитратною з погляду обсягів інформації та вимог до обчислювальної потужності. Проведений ROC аналіз показав високу якість розробленого нечіткого класифікатора.

Ключові слова: нечіткий класифікатор зображень; велика динамічна база даних; спотворені версії зображень; комп'ютерна система; швидкий пошук.