Article

Dual-ISM: Duality-Based Image Sequence Matching for Similar Image Search

Hye-Jin Lee ¹, Yongjin Kwon ² and Sun-Young Ihm ^{3,*}

¹ Engineering Research Center, Pai Chai University, 155-40 Baejae-ro, Seo-gu, Daejeon 35345, Korea; hyejinlee@pcu.ac.kr

² Artificial Intelligence Research Laboratory, Electronics and Telecommunications Research Institute, 218 Gajeong-ro, Yuseong-gu, Daejeon 34129, Korea; scocso@etri.re.kr

³ Department of Computer Engineering, Pai Chai University, 155-40 Baejae-ro, Seo-gu, Daejeon 35345, Korea

* Correspondence: sunnyihm@pcu.ac.kr

Abstract: In this paper, we propose the duality-based image sequence matching method, which is called Dual-ISM, a subsequence matching method for searching for similar images. We first extract feature points from the given image data and configure the feature vectors as one data sequence. Next, the feature vectors are configured in the form of a disjoint window, and a low-dimensional transformation is carried out. Subsequently, the query image that is entered to construct the candidate set is similarly subjected to a low-dimensional transformation, and the low-dimensional transformed window of the data sequence and window that are less than the allowable value, ϵ , is regarded as the candidate set using a distance calculation. Finally, similar images are searched in the candidate set using the distance calculation that are based on the original feature vector.

Keywords: similar image search; image sequence matching; subsequence matching

Citation: Lee, H.-J.; Kwon, Y.; Ihm, S.-Y. Dual-ISM: Duality-Based Image Sequence Matching for Similar Image Search. *Appl. Sci.* **2022**, *12*, 1609. <https://doi.org/10.3390/app12031609>

Academic Editors: Jinho Kim and Rubén Usamentiaga

Received: 30 November 2021

Accepted: 30 January 2022

Published: 3 February 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Driven by the rapid growth of image and video data, object recognition, and large-scale image searching in real-time videos have received a great deal of attention [1]. These technologies are developing into a variety of applications, such as monitoring and tracking criminal behaviors and continuously tracking specific objects. To search for or track objects in a video or in a live stream, accurate object recognition is required. Hence, image similarity search technology is required to determine whether items across different frames are the same object or target class [2]. Image search techniques are being actively studied in various fields, such as content-based [3] and hash-based image searching [4]. With the recent development of image-related deep learning technologies, many image and video-related deep learning-based studies are being conducted such as AlexNet [5], R-CNN [6], Faster-R-CNN [7], and MATNet [8]. However, doing so with deep learning tools (e.g., convolutional neural networks (CNNs)) requires vast amounts of time for repetitive learning and similarity measurements because all the pixels of an image are searched to determine the similarities. Also, deep learning-based image classification aims to classify the class of images, and if there is no prior class information or the class is different, it is not classified as a similar image. The search for image similarities is based on the extracted object features. As machine learning technologies are developed, many studies on feature extraction from images are being accomplished. Object features are values that can be used to distinguish one object from another (e.g., color, illuminance, shape, texture, and construction). Features provided in these ways are gradually diversified for accurate object detection, gradually becoming higher-dimensional.

This paper proposes the duality-based image sequence matching (Dual-ISM) method, a subsequence matching method for searching for similar images. For image classification and searching, a feature vector must first be extracted from the image because image sizes

are normally quite large. However, even if the feature vectors can be extracted, there are an excessive number for comparative operations. Therefore, an efficient method is required to search for an image based on large-capacity, high-dimensional feature data. Time-series data reflect features whose values change over time (e.g., stocks, music, and weather). Finding similar sequences among user queries enables sequence matching within the time-series data. Based on this knowledge, this paper proposes a method that uses sequence matching techniques to search for similar images based on image vector data. A sequence matching method is proposed for efficient image searching that is based on high-dimensional feature vectors. The proposed method first extracts the feature points of the given image data and constructs a feature vector, which is then configured into a disjoint window. Subsequently, a low-dimensional transformation is performed for each window, and a candidate set is constructed using a distance comparison with a query image sequence. During this process, the query image is similarly subjected to a low-dimensional transformation to calculate the distance. Finally, for the candidate set, similar images are searched by calculating the actual distance between the original feature vectors.

The remainder of this paper is organized as follows. Section 2 describes the related research and Section 3 describes the proposed sequence matching method. Section 4 demonstrates the excellence of the proposed method through various experiments. Finally, Section 5 presents the conclusions of this paper and recommends future studies.

2. Related Work

This section describes extant studies that are related to the proposed method. First, Section 2.1 introduces subsequence matching studies and Section 2.2 introduces low-dimensional transformation techniques.

2.1. Sequence Matching Methods

In this section, sequence matching studies in time series data are first described. Sequence matching when the lengths of the data and query sequences are not the same is called subsequence matching. In subsequence matching, when the query sequence and the allowable value, t , are given, all the similar sequences in which the distance from the query sequence is less than t in the data sequence are obtained. In this case, because the length of the sequence is long, the distance is calculated by dividing the sequence into windows of appropriate sizes. When the length of the window is w , the method of dividing the index of the window by increasing it by one is called a sliding window, and the method of dividing the index of the window by increasing it by w is called a disjoint window. The I-adaptive method [9] obtains a similar sequence by dividing the data sequence into a sliding window and dividing the query sequence into a disjoint window. This method has a problem in that the similar sequence candidate set becomes quite large. Meanwhile, dual matching [10] divides a data sequence into a disjoint window and a query sequence into a sliding window. It divides a subsequence of a data sequence into a window, and if the length of the window does not match the size of the subsequence, it cannot be configured as a window; hence, a remainder data value is generated. E-dual matching [11] configures data that cannot be compared because it cannot be configured as a window during query processing as a virtual window. Furthermore, it cannot be utilized to determine similar sequences. ChainLink [12] is a lightweight distributed indexing framework that supports k -nearest neighbor queries, constructs a subsequence that is based on dual matching, and uses hashing techniques for subsequence matching. L-matching [13] applies a query processing method using two-level-based indices and lower-bound-based pruning for efficient subsequence matching. KV-matching [14] provides a single index that supports various queries and performs query processing with several index sequential scans. It can be easily implemented in local files or in Hbase tables with key-value structures. Mueen et al. proposed MASS, an algorithm that creates a query distance profile for long time series [15]. MASS uses a convolution-based method to calculate sliding dot products in $(n \log n)$ time. It also applies a just-in-time z -normalization technique. We use a convolution to compute all the sliding

dot products between the query and the sliding windows. Here, for the convolution, if x and y are vectors of polynomial coefficients, convolving them is equivalent to multiplying the two polynomials.

2.2. Low-Dimensional Transformation Techniques

This section describes a dimension reduction method that reduces high-dimensional to low-dimensional data. The representative dimension reduction algorithms include principal component analysis (PCA), linear discriminant analysis (LDA), and non-negative matrix factorization (NMF).

The PCA method generates a new main component that is represented by a linear combination of variables through the covariance and correlation matrices of variables. It also refers to a dimension reduction method in which the total variance of the variables can be explained through the main component. The PCA method tends to find a projection matrix, W_{opt} , that maximizes the coefficient of determination of the total scatter matrix [16].

$$W_{opt} = \underset{w}{argmax} |W^T S_T W|, \quad (1)$$

where S_T represents all the scatter matrices and leads to the following equation:

$$S_T = \sum_{i=1}^M (x_i - \mu)(x_i - \mu)^T, \quad (2)$$

where μ represents the average shape vector of all the samples in the training set, x_i represents the shape vector of the i -th sample, and M represents the total number of training samples.

The LDA method is a dimension reduction technique that is used in classification problems [17]. The method aims to optimize classifications for given classes by maximizing the ratio of between-class to within-class variances of a given dataset [18]. Here, the key is to separate the class mean in the projected direction while achieving a small variance around the means. Through this method, LDA efficiently reduces multidimensional data into low-dimensional spaces; hence, it is suitable for the graphical representation of datasets [19]. In other words, the LDA method attempts to solve the optimal discrimination projection matrix, W_{opt} , using the following equation [12]:

$$W_{opt} = \underset{w}{argmax} \frac{|W^T S_B W|}{|W^T S_W W|}, \quad (3)$$

where S_b is an interclass scatter matrix and S_W is an intra-class scatter matrix.

$$S_b = \sum_{i=1}^J (\mu_i - \mu)(\mu_i - \mu)^T, \quad (4)$$

$$S_W = \sum_i (x_i - \mu_{k_i})(x_i - \mu_{k_i})^T, \quad (5)$$

where J represents the total number of samples in the entire image set, μ_i is the average shape vector of image class i , and n_i is the number of samples of image class i . Further, x_i denotes a feature vector for a sample, and μ_{k_i} denotes a vector of an image class to which x_i belongs.

The NMF [20] is a matrix factorization algorithm that focuses on the analysis of data matrices that are composed of non-negative components. In this case, both the main component and the coefficient must be greater than or equal to zero, and if they are positive, the features can be divided without the superiority or inferiority of the components. The NMF aims to find two non-negative matrices, $U = [u_{ij}] \in \mathbb{R}^{m \times k}$ and $V = [v_{ij}] \in \mathbb{R}^{n \times k}$, that minimize the following objective function [21]:

$$O = \|X - UV^T\|_F^2, \quad (6)$$

where each column of X is a sample vector when X , the data matrix, is $X = [x_1, x_2, \dots, x_n] \in \mathbb{R}^{m \times n}$. Qian et al. [22] proposes a matrix-vector nonnegative tensor factorization (NTF) model, which is an extended method from NMF-based models.

3. Dual-ISM (Proposed Method)

This section describes the duality-based image sequence matching (Dual-ISM) method that is proposed in this study. It is a subsequence matching method to search for similar images. The proposed Dual-ISM method consists of the following four steps, and the details for each are described in Sections 3.1–3.4, respectively.

- Step 1: Feature point extraction; feature points are extracted from the original image, and a feature vector is constructed.
- Step 2: Low-dimensional transformation; for efficient subsequence matching, feature vectors that are extracted from the image are configured as disjoint windows and low-dimensional transformation is performed.
- Step 3: Candidate set construction; a set of candidates for subsequence matching is configured with windows in which the distance between the low-dimensional transformed data sequence and the query sequence is less than a given allowable value, ϵ .
- Step 4: Searching for similar image sequences; similar image sequences are searched for the constructed candidate set using a distance calculation that is based on the original feature vector.

Figure 1 shows the implementation process of the proposed method. First, feature points are extracted from the given image data and the feature vectors are then configured as one data sequence. Next, feature vectors are configured in the form of a disjoint window and a low-dimensional transformation is carried out. Subsequently, the query image that is entered to construct the candidate set is similarly subjected to a low-dimensional transformation, and the low-dimensional transformed window of the data sequence and window less than the allowable value, ϵ , is regarded as the candidate set using a distance calculation. Finally, similar images are searched in the candidate set using the distance calculation that is based on the original feature vector.

Figure 1. Overall Dual-ISM process.

3.1. Step 1: Feature Point Extraction

In this subsection, the process of detecting feature points from images and extracting the feature quantities using the KAZE algorithm is described. A total of 128 feature

vectors are extracted and constructed by detecting the feature points from the original two-dimensional (2D) image. During this process, the feature point is extracted using KAZE, which detects the feature point by finding normalized Hessian local maxima of different scales. The KAZE algorithm was proposed as a way of using a non-linear scale space with a non-linear diffusion filter [23]. This has the advantage of making the blurring of an image locally adapt to the feature point; it reduces the noise and simultaneously maintains the area boundary of the target image [24]. The KAZE detector is based on the scale normalization determinant of the Hessian matrix. The KAZE function does not change in rotation, scale, or limited affine, but it has more differentiation at various scales according to the increase in computational time. Equation (7) represents a standard nonlinear diffusion equation.

$$\frac{\partial L}{\partial t} = \text{div}(c(x, y, t) \cdot \nabla L), \quad (7)$$

where c is the conductivity function, div is divergence, ∇ is the gradient operator, and L is image luminance. The Hessian coefficient of determination is calculated for each filtered image, L_i , in a nonlinear scale space as follows [25]:

$$L_{Hessian}^i = \sigma_{i,norm}^4 (L_{xx}^i L_{yy}^i - L_{xy}^i L_{xy}^i), \quad (8)$$

where $\sigma_{i,norm}$ represents a normalized scale coefficient and L_{xx} , L_{yy} , and L_{xy} refer to the secondary derivatives in horizontal, vertical, and intersecting directions, respectively [26]. The feature point that is detected through KAZE in this manner is shown in Figure 2, and the extracted feature vector appears as one sequence, as presented in Figure 3.

Figure 2. Feature point extraction results using the KAZE algorithm. (a) Flowers-17 (Daffodil); (b) ILSVRC2012 (Pineapple).

3.2. Step 2: Low-Dimensional Transformation

In this subsection, the low-dimensional transformation step is described. For efficient subsequence matching, a low-dimensional transformation is performed to reduce the dimensions on feature vectors that are extracted from a single 2D image. The feature points composed of one sequence that are extracted in Step 1 are composed of n subsequences with w feature vectors. After configuring this in the form of a disjoint window, the dimensions of each window are reduced using the LDA, NMF, and PCA methods for low-dimensional transformation in this study. For image search, the size of the window is set to w , which is equal to the number of feature vectors.

Figure 3. Example of an image feature point vector. (a) Flowers-10 (Daffodil); (b) ILSVRC (pineapple).

3.3. Step 3: Candidate Set Construction

In this subsection, the process of constructing a candidate set is described. For subsequence matching, the candidate set is constructed using sequences in which the distance between the query sequence and the low-dimensional transformed data sequence in Step 2 is less than a given allowance value, ε . The calculation of the distance between the query sequence and the low-dimensional transformed data sequence uses the Euclidean distance, d , and is calculated as outlined in Equation (9):

$$d(x, y) = \sqrt{\sum_{i=1}^n (y_i - x_i)^2} \quad (9)$$

3.4. Step 4: Searching for Similar Image Sequences

In this subsection, the method of searching for a similar image sequence is described. A similar image sequence is searched by calculating the distance from the query sequence based on the original feature vector for the candidate set that is constructed in Step 3. In this case, the distance calculation for the candidate set consisting of a query sequence and sequences that are smaller than the allowable value, ε , is performed using the Euclidean distance calculation in the same manner as Step 3. Afterward, the Euclidean distance, d , of the query sequence and candidate set are arranged in ascending order. A similar image sequence for the query sequence is searched and extracted from the candidate set based on the sorted dataset.

4. Experimental Results

This section introduces the experimental environment and the test results of the proposed method. First, the dataset that was used in the experiment is described. In the experiment, the Flower dataset consisted of 17 flower variety classes and 80 images per class [27]. The second dataset (i.e., ILSVRC) had 1000 classes [28]. In this study, some data from the published Flower dataset and ILSVRC were used in the experiment. For the

experiment, we set the size of the window as the size of the image vector and the allowable value, ϵ , was set such that the top 30 images were extracted.

4.1. Comparative Experiment of Similar Image Search Accuracy—Low-Dimensional Transformation Methods

In this experiment, a comparative experiment of similar image searching accuracy for the proposed Dual-ISM method was conducted. In this paper, we define accuracy as the probability of including an image with the same class among images that were determined as similar images, and the equation is as follows.

$$\text{accuracy} = \frac{\text{the number of images with same class}}{\text{the number of similar iamges}} \quad (10)$$

The methods were compared for searching for similar images as follows:

- 128 d+ED: Euclidean distance calculation that is based on the original feature vector
- LDA+ED: Euclidean distance calculation after LDA low-dimensional transformation
- Dual-ISM (LDA): the proposed method using LDA low-dimensional transformation
- PCA+Dual-ISM (PCA): the proposed method using PCA low-dimensional transformation
- NMF+Dual-ISM (NMF): the proposed method using NMF low-dimensional transformation

As a result, as shown in Figure 4a, for the Flowers-10 dataset, two methods, LDA+ED and Dual-ISM (LDA), found eight images such as the query sequence and searched for most similar images. In the case of the ILSVRC dataset in Figure 4b, the two methods similarly recorded the highest accuracy. Based on the experimental results, LDA recorded the highest accuracy among the three low-dimensional transformation methods.

Figure 4. Comparison of similar image search accuracy. (a) Accuracy comparison for the Flowers dataset and (b) the accuracy comparison for the ILSVRC dataset.

Although the two methods had the same number of similar images that were searched, there was, however, a difference in the real distance between the sequences. In this paper, we define precision by scoring the distance ranking of images of the same class among similar images and the equation is as follows. Here n is the number of similar images and the rank score was calculated higher as it was closer to the query image sequence.

$$\text{precision} = \frac{\sum \text{the rank score of same class image}}{\sum_{i=1}^n i} \quad (11)$$

As the result, the precision of the Dual-ISM was 45% and LDA+ED was 23%, and the actual distance calculation results are shown in Tables 1 and 2. The lines that are shaded in Tables 1 and 2 are image sequences that have the same class among the results of searching

for similar images to the query image. Table 1 shows a list of the search results of LDA + ED, and Table 2 shows a list of the search results of the proposed Dual-ISM (LDA) method. It is a search result for a similar image of the final top 30, and the image displayed in green is an image with the same class as the query image. In the case of the LDA+ED method, eight images that had the same class as the query image were derived as a result, but when comparing only the top 10, the accuracy was 20%. However, it was identified that, when using the proposed method, the same classes as the query image were searched at the top in the result.

Table 1. Results of searching for similar images. (LDA+ED).

Class	Component1	Component2	...	Component8	Actual Distance	Rank Score
0	0.4148	0.8275	...	0.7073	0.95332	30
6	1.1050	−0.3521	...	0.9043	1.87587	
0	−0.5677	1.0200	...	−0.0351	2.07946	28
6	0.5831	0.0016	...	1.0581	2.11729	
3	0.4737	0.8336	...	−0.2374	2.21694	
6	−0.6978	1.1347	...	0.5265	2.22265	
6	−0.4595	0.3928	...	0.6221	2.24890	
3	0.2888	0.7531	...	−0.3324	2.29725	
...	...	...	...	...	...	...
0	0.1168	0.9978	...	0.1012	2.57962	1

Table 2. Results of searching for similar images. (Dual-ISM).

Class	Component1	Component2	...	Component128	Actual Distance	Rank Score
0	0.0010	−0.0074	...	0.0590	1.31587	30
0	0.0012	0.0016	...	0.0609	1.36660	29
0	0.0008	0.0036	...	0.0655	1.41247	28
0	−0.0096	−0.0198	...	0.1271	1.47892	27
0	−0.0226	0.0163	...	0.0195	1.52239	26
0	−0.0475	0.0212	...	0.0300	1.57971	25
0	0.0456	0.0255	...	0.1115	1.60463	24
0	0.0411	0.0069	...	0.0980	1.611103	23
...	...	...	...	...	...	...
9	−0.0002	0.0004	...	0.0129	9.13522	

4.2. Comparative Experiment of Similar Image Search Accuracy—Subsequence Matching

In this experiment, the accuracy of searching for similar images of the proposed Dual-ISM method was compared to that of the existing subsequence matching method, Mass-ts. As the proposed method showed the best accuracy when using the LDA transformation, comparisons were conducted with Dual-ISM (LDA) after performing all the low-dimensional transformation methods on the Mass-ts method.

Based on the experimental results, in the case of the Flowers-10 dataset in Figure 5a, the two methods of Mass-ts (LDA) and Dual-ISM (LDA) found eight images that were similar to the query sequence and searched for the most similar images. In terms of the ILSVRC dataset in Figure 5b, the proposed method recorded the highest accuracy, and Mass-ts (LDA) recorded the second highest.

Figure 5. Comparison of similar image search accuracy. (a) Accuracy comparison for the Flowers dataset and (b) the accuracy comparison for the ILSVRC dataset.

4.3. Comparison of Similar Image Search Results

Figures 6 and 7 compare the sequences for the top three results of a similar image search and the actual image. Figure 7 shows the top three sequences, showing the sequence of the query image at the top and the sequence of the top three images because of searching for similar images using the Dual-ISM and 128 d+ED methods.

Figure 6. Similar image search results (top-three image sequences).

Figure 7. Similar image search results (actual top-three images).

Figure 7 shows the actual image results. A query image is given at the top of the table, and among the methods that were used in the experiment, four methods that had satisfactory results were compared. In the proposed method, only images of the same class as the actual image were searched, and, when comparing with the actual images, it was confirmed that they were similar. The remaining three methods show that images with different classes from the query image were derived from the top-three results, and the actual images were also significantly different.

4.4. Comparison of Similar Image Search Speeds

In this experiment, for each dataset, a similar image search speed according to a combination of a low-dimensional transformation technique and a query sequence was compared. Figure 8 shows the results of the comparison of similar image search speeds between Dual-ISM(LDA), Mass-ts(LDA) and LDA+ED. The proposed method takes a longer search speed than LDA+ED, but shows high precision. And it shows better results in both search speed and accuracy than Mass-ts(LDA).

Figure 8. Comparison of similar image search speeds.

5. Conclusions

In this paper, Dual-ISM, which matches image sequences based on duality, was proposed for efficient similar image searching. Dual-ISM first constructs a feature vector by extracting the feature points from images that are given through the KAZE algorithm. Next, the feature vectors were constructed as data sequences, configured in the form of disjoint windows. Then, dimensions were reduced through low-dimensional transformation techniques (i.e., LDA, PCA, and NMF). The query image was subjected to low-dimensional transformation after extracting the feature points, and a candidate set was constructed through a distance comparison with the window of the low-dimensional transformed data sequence. Finally, for the candidate set, similar images were searched through the calculation of the actual distance between the original feature vectors. Based on the experimental results, among the low-dimensional transformation techniques, the LDA method reduced the dimension to the form maintaining the class best, and Dual-ISM showed the highest accuracy in similar image search results. In an experiment deriving the top 30, conventional Mass-ts and Euclidean distance calculation methods showed similar high accuracy; however, when comparing the final image search results with actual distance calculations, it was confirmed that the proposed method most accurately searched for images that were similar to the query image in order. As for future work, we will study efficient feature-point extraction methods to support similar image searches for large-capacity image datasets. Also, we will compare these with deep learning-based methods and review applying the proposed method as a preprocessing method of a deep learning model.

Author Contributions: Conceptualization, Y.K.; Methodology, S.-Y.I.; Software, H.-J.L.; Writing—original draft, H.-J.L.; Writing—review & editing, S.-Y.I. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No. 2020-0-00004, Development of Previsional Intelligence based on Long-term Visual Memory Network).

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Lu, J.; Liong, V.E.; Zhou, J. Deep Hashing for Scalable Image Search. *IEEE Trans. Image Process.* **2017**, *26*, 2352–2367. [[CrossRef](#)] [[PubMed](#)]
2. Choi, D.; Park, S.; Dong, K.; Wee, J.; Lee, H.; Lim, J.; Bok, K.; Yoo, J. An efficient distributed in-memory high-dimensional indexing scheme for content-based retrieval in spark environments. *J. KIISE* **2020**, *47*, 95–108. [[CrossRef](#)]
3. Shao, Y.; Mao, J.; Liu, Y.; Zhang, M.; Ma, S. Towards context-aware evaluation for image search. In Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, Paris, France, 21–25 July 2019; pp. 1209–1212.
4. Deng, C.; Yang, E.; Liu, T.; Li, J.; Liu, W.; Tao, D. Unsupervised Semantic-Preserving Adversarial Hashing for Image Search. *IEEE Trans. Image Process.* **2019**, *28*, 4032–4044. [[CrossRef](#)] [[PubMed](#)]
5. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet classification with deep convolutional neural networks. *Adv. Neural Inf. Process. Syst.* **2012**, *25*, 1097–1105. [[CrossRef](#)]
6. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. In Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 580–587. [[CrossRef](#)]
7. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. *Adv. Neural Inf. Process. Syst.* **2015**, *28*, 91–99. [[CrossRef](#)] [[PubMed](#)]
8. Zhou, T.; Li, J.; Wang, S.; Tao, R.; Shen, J. MATNet: Motion-Attentive Transition Network for Zero-Shot Video Object Segmentation. *IEEE Trans. Image Process.* **2020**, *29*, 8326–8338. [[CrossRef](#)] [[PubMed](#)]
9. Faloutsos, C.; Ranganathan, M.; Manolopoulos, Y. Fast subsequence matching in time-series databases. *ACM SIGMOD Rec.* **1994**, *23*, 419–429. [[CrossRef](#)]
10. Moon, Y.S.; Whang, K.Y.; Loh, W.K. Duality-based subsequence matching in time-series databases. In Proceedings of the 17th International Conference on Data Engineering, Heidelberg, Germany, 2–6 April 2001; pp. 263–272.

11. Ihm, S.-Y.; Nasridinov, A.; Lee, J.-H.; Park, Y.-H. Efficient duality-based subsequent matching on time-series data in green computing. *J. Supercomput.* **2014**, *69*, 1039–1053. [[CrossRef](#)]
12. Alghamdi, N.; Zhang, L.; Zhang, H.; Rundensteiner, E.A.; Eltabakh, M.Y. ChainLink: Indexing Big Time Series Data for Long Subsequence Matching. In Proceedings of the 2020 IEEE 36th International Conference on Data Engineering (ICDE), Dallas, TX, USA, 20–24 April 2020; pp. 529–540. [[CrossRef](#)]
13. Feng, K.; Wang, P.; Wu, J.; Wang, W. L-Match: A Lightweight and Effective Subsequence Matching Approach. *IEEE Access* **2020**, *8*, 71572–71583. [[CrossRef](#)]
14. Wu, J.; Wang, P.; Pan, N.; Wang, C.; Wang, W.; Wang, J. KV-Match: A Subsequence Matching Approach Supporting Normalization and Time Warping. In Proceedings of the 2019 IEEE 35th International Conference on Data Engineering (ICDE), Macao, China, 8–11 April 2019; pp. 866–877. [[CrossRef](#)]
15. Mueen, A.; Zhu, Y.; Yeh, M.; Kamgar, K.; Viswanathan, K.; Gupta, C.K.; Keogh, E. The Fastest Similarity Search Algorithm for Time Series Subsequences under EUCLIDEAN Distance. 2015. Available online: <http://www.cs.unm.edu/~mueen/FastestSimilaritySearch.html> (accessed on 30 November 2021).
16. Ye, F.; Shi, Z.; Shi, Z. A comparative study of PCA, LDA and Kernel LDA for image classification. In Proceedings of the International Symposium on Ubiquitous Virtual Reality, Guangju, Korea, 8–11 July 2009; pp. 51–54.
17. Belhumeur, P.; Hespanha, J.; Kriegman, D. Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection. *IEEE Trans. Pattern Anal. Mach. Intell.* **1997**, *19*, 711–720. [[CrossRef](#)]
18. Jafri, R.; Arabnia, H.R. A Survey of Face Recognition Techniques. *J. Inf. Process. Syst.* **2009**, *5*, 41–68. [[CrossRef](#)]
19. Karg, M.; Jenke, R.; Seiberl, W.; Kühnlenz, K.; Schwirtz, A.; Buss, M. A comparison of PCA, KPCA and LDA for feature extraction to recognize affect in gait kinematics. In Proceedings of the 3rd International Conference on Affective Computing and Intelligent Interaction and Workshops, Amsterdam, The Netherlands, 10–12 September 2009; pp. 1–6.
20. Lee, D.D.; Seung, H.S. Learning the parts of objects by non-negative matrix factorization. *Nature* **1999**, *401*, 788–791. [[CrossRef](#)] [[PubMed](#)]
21. Cai, D.; He, X.; Wu, X.; Han, J. Non-negative Matrix Factorization on Manifold. In Proceedings of the 2008 Eighth IEEE International Conference on Data Mining, Pisa, Italy, 15–19 December 2008; pp. 63–72. [[CrossRef](#)]
22. Qian, Y.; Xiong, F.; Zeng, S.; Zhou, J.; Tang, Y.Y. Matrix-Vector Nonnegative Tensor Factorization for Blind Unmixing of Hyperspectral Imagery. *IEEE Trans. Geosci. Remote Sens.* **2017**, *55*, 1776–1792. [[CrossRef](#)]
23. Alcantarilla, P.F.; Bartoli, A.; Davison, A.J. KAZE features. In Proceedings of the European Conference on Computer Vision, Florence, Italy, 7–13 October 2012; pp. 214–227.
24. Tareen, S.A.K.; Saleem, Z. A comparative analysis of SIFT, SURF, KAZE, AKAZE, ORB, and BRISK. In Proceedings of the 2018 International Conference on Computing, Mathematics and Engineering Technologies (iCoMET), Sukkur, Pakistan, 3–4 March 2018; pp. 1–10.
25. Ma, X.; Xie, Q.; Kong, X. Improving KAZE feature matching algorithm with alternative image gray method. In Proceedings of the 2nd International Conference on Computer Science and Application Engineering, Hohhot, China, 22–24 October 2018; pp. 1–5.
26. Demchev, D.; Volkov, V.; Kazakov, E.; Alcantarilla, P.F.; Sandven, S.; Khmeleva, V. Sea Ice Drift Tracking from Sequential SAR Images Using Accelerated-KAZE Features. *IEEE Trans. Geosci. Remote Sens.* **2017**, *55*, 5174–5184. [[CrossRef](#)]
27. Nilsback, M.-E.; Zisserman, A. A Visual Vocabulary for Flower Classification. In Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06), New York, NY, USA, 17–22 June 2006; pp. 1447–1454. [[CrossRef](#)]
28. Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M.; et al. Imagenet large scale visual recognition challenge. *Int. J. Comput. Vis.* **2015**, *115*, 211–252. [[CrossRef](#)]