Content-based image similarity measurement grounded on information retrieved by semantic segmentation algorithms

R. WASILUK

rafal.wasiluk@student.wat.edu.pl

Military University of Technology, Doctoral School
Kaliskiego Str. 2, 00-908 Warsaw, Poland

The purpose of this article is to present a novel approach for recording information contained in an image in a structured form and performing image similarity assessment with use of these data structures. The solution presented in this document relies on an analysis of results produced by pre-trained semantic segmentation algorithms. These outcomes can be transformed to a set of vectors representing some characteristics of each class of objects detected in the provided image. These data structures can contain meaningful information about algorithm detections, such as the object's position on the image, the object's size compared to the overall image size or the object's dominant colors, etc. Vectors prepared as described previously can be further compared with other image embeddings using many mathematical tools like distance measures. Moreover, the approach described in this article allows the user to define a value of weight tied to each characteristic. This provides the ability to make a subset of features more important than others and have a greater impact on the final value of image similarity.

Keywords: image similarity, semantic segmentation, vector image representation.

DOI: 10.5604/01.3001.0055.0855

1. Introduction

Comparing objects is a fundamental operation in human life. It is used subconsciously several times a day in many areas of life, such as shopping in a grocery store, choosing the best path to reach the destination of a journey, etc. All the mentioned situations have many possible choices. Furthermore, each can be described by a set of verbal and numerical features. If the data representing objects is provided to the decision maker in a numerical form, then this task is easy to perform by both a man and a machine. However, if the number of considered features grows then the described process is getting more and more confusing for people and only a little bit more difficult for computers using appropriate optimization algorithms. The situation is significantly different when the input data for the object comparing process is provided in the form of images. Then the difficulty of the task grows significantly for a computer and stays almost at the same level for a human. This is because computers are not able to see what kind of objects are presented in images and they have no ability to describe those objects with a set of characteristics. To address this issue, computers must use algorithms to process graphical data. They can be very helpful in extracting an object's features like shape, color, texture, and numerical characteristics. Moreover, they can be helpful with assessing values of features tied with graphic frames like vertical and horizontal object's position.

Despite the difficulties of dealing with graphical data, people are constantly trying to automate processes of image similarity assessment and similar image retrieval. Recently, the development of graphical data processing techniques, using artificial intelligence tools like semantic segmentation algorithms, has created opportunities for retrieving meaningful information from images [1]. Therefore, the logical consequence of these inventions is to use them along with computers to assess image similarity.

However, defining image similarity could be a challenging task. This is because different decision-makers could understand the same object similarity assessment task differently. For example, if a human is comparing two objects that belong to the same class, he may evaluate their similarity based on various characteristics like shape, color, size, or texture [2]. Every time the most important feature may vary depending on

the context, the decision maker's individual preferences, the task to be performed, and its goal which has to be achieved.

So, to automatically perform the task of assessing image similarity there is a need for an efficient way to transform images to the form of data that can be easily interpreted by computers. There is also the necessity of taking into consideration decision-maker preferences.

To address the described issues, this article will propose an approach to express and store what is presented in an image in computer-understandable form. Furthermore, it will also suggest a method of comparing these representations according to the decision-maker's preferences.

2. Related works

Research on image similarity measures and techniques has been conducted by researchers for decades [3]. There are a lot of existing approaches to perform graphics comparisons and image retrieval from huge databases [2], [4], [5], [6], [7]. So, this section of the article will present some of the existing methods of image comparison.

The first method of image comparison which will be described in this section is based on human-assigned annotations. Each of the graphics presented to the algorithm is initially provided with a set of words or phrases describing depicted objects or situations. Then an intersection of annotation sets is performed. If the resulting subset is close to the initial sets, it means that the images are alike. Systems utilizing keyword-based methods of similar picture retrieval are often called by the acronym TBIR [4], [6] which stands for Text-Based Image Retrieval. Although the algorithm used by these pipelines is simple, it has a few significant drawbacks. Firstly, the process of manual image annotation with keywords is very time-consuming, especially for a widespread set of provided graphical data. Moreover, this method is subjective because it is based on human perception of the world. Thus, it is very likely that the same image will be described by two people with different sets of annotation keywords [4]. So, this approach could not be efficiently used in real-world cases[1].

The method of image similarity assessment which is presented in the previous paragraph is semi-automated and primitive. It requires a human to perform a labeling process for each image to determine the contents of the graphic. That information allows a pair of images to be successfully compared with each other. Of course, this method has a huge bottleneck. It is a manual image labeling. Automating this part of the pipeline can be achieved by using object detection or classification algorithms. The mentioned methods use pre-trained artificial intelligence models to specify what is presented in the image [1]. An example of this system could be the YOLO model [8]. It was presented to the scientific audience by a group of researchers in 2016. And now after a few years, it is still one of the state-of-the-art approaches to perform object detection in images. It uses a single neural network to predict multiple objects bounding boxes and classes from just one evaluation of the algorithm. Retrieved information about the presence of objects representing pre-trained classes in images can be further utilized by algorithms for image comparison. Simply, as in the previous method, detected object names or assigned numbers are treated as labels. Although the presented method is automated, it has also a lot of significant drawbacks. The main one certainly is the lack of ability to retrieve other objects' features than their presence or absence in the image. Usage of this method can lead to omitting information about an object's shape, color, or texture.

One of the first automated attempts to assess image similarity with computers and Deep Neural Networks (DNN) was presented by researchers from AT&T Bell Laboratories [7]. They proposed to use a new neural network architecture (called "Siamese") to perform a human signature verification between the stored ground truth picture of a signature and a provided image of a doubtful duplicate of it. Going into details, the solution was created by merging outcomes of two identical pre-trained neural networks (equipped with the same set of layers and weights) into a shared distance layer, which was able to calculate the difference between feature vectors of provided images. The doubtful signature is considered similar to ground truth if the value of distance measure between picture embeddings is lower than the pre-defined threshold value. This method is very useful when a decision-maker wants to assess the similarity of plain objects like human signatures or frontal images of human faces. The biggest disadvantage of such solutions is that they require images presented to the Siamese model to be preprocessed to optimize the output quality. The reason for this is that this algorithm considers full-size input images. So even a small shift of object in the picture frame, a change of the background, or even a change of some pixels may lead to a completely different outcome [9].

Nowadays CBIR (Content-Based Image Retrieval) systems are becoming more and more popular due to the significant growth of images stored in databases on servers and a notable increase in the number of images sent over the Internet. Such systems are progressively being used for fetching alike images from a wide collection of them [4]. They are doing it by performing three steps: extracting features, indexing them, and retrieving similar graphical data. In the context of this paper, all stages of the CBIR systems pipeline are worth inspecting. The first of them is responsible for identifying visual features depicted in graphics. It retrieves low-level features like edges, colors, contours, angles, or textures along with high-level features that identify more complex structures like human faces, car wheels, etc. To perform this task, CBIR systems usually use tools such as Convolutional Neural Networks (CNN), color histograms, and wavelet transforms. The second of the pipeline stages oversees converting outcomes from the previous step to the form, in which the result data can be stored in some kind of database system. The following tools are used to record data efficiently for future high-speed retrieval and reading: vector quantization, clustering, and tree-based indexing. Finally, the output image representations are provided to the third stage of the CBIR pipeline. It is tasked with retrieving similar images to the query picture based on embeddings previously stored in the database. At this stage, a variety of algorithms are utilized to evaluate similarity. The simplest ones, used for this task are Euclidean distance, cosine similarity, and correlation coefficient [10]. The method described in this paragraph is achieving very good efficiency, but it is still not perfect. It still omits certain types of information, such as the position of the object, which could be crucial in some cases.

3. Suggested approach

According to the best knowledge, there were no previous attempts to perform image similarity assessment with the use of semantic segmentation algorithms. So, in the following section of the article, a new approach to graphical likeness measurement will be presented.

The suggested approach will be divided into a chain of consecutive steps. Each of them will be responsible for performing a different task on incoming data. The first of them is responsible for extracting objects and their features from the provided image. The next one will be tasked with processing the outcome from the previous step and forming the received data into a matrix representation. Finally, an embedding produced for the input image will be compared to other image representations produced by the algorithm. This step will return information about the provided image similarity. The diagram of the suggested image similarity assessment solution is presented in Figure 1.

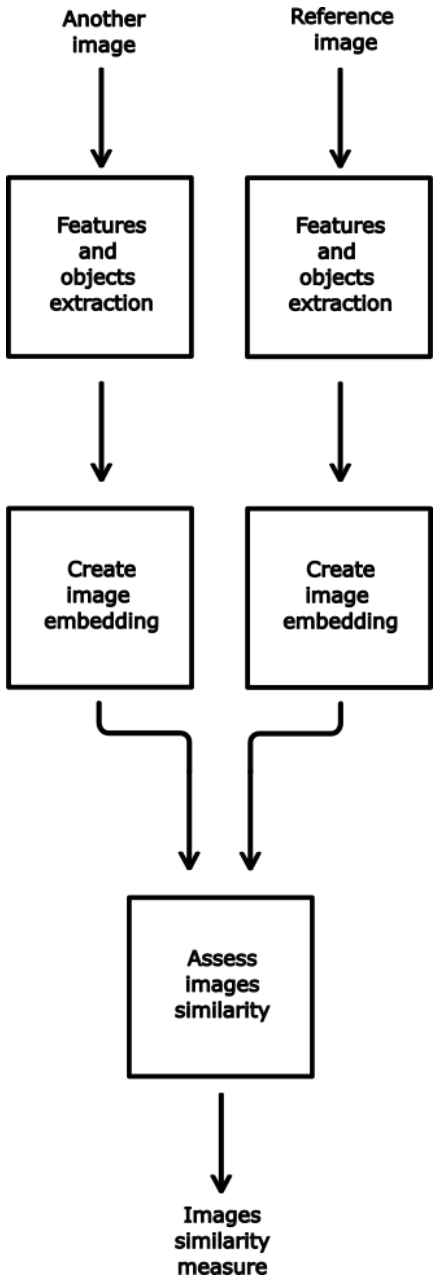

Fig. 1. Model of the suggested image similarity assessment method

The first stage of image similarity assessment in the suggested approach is tasked with extracting objects and their features from an input image. In the presented approach a semantic segmentation algorithm will be responsible for performing this task. It can recognize presented objects, and it can also retrieve detected objects'

positions in the provided image. It is possible because the outcome of image segmentation algorithms is not only binary information (the image has the object of the given class or not), but it also provides information about the object's location in the image. This position is returned as a subset of the original image pixels, which contain a recognized object instance. An example outcome from the semantic segmentation algorithm is visualized in Figure 2.

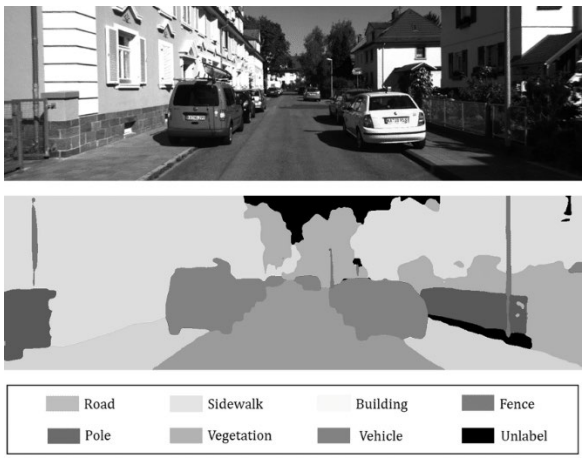

Fig. 2. An example of image semantic segmentation. The upper graphic is the input image and the lower one is the outcome

As we can see, the image provided to the algorithm (the upper internal graphic) was covered with masks (the result is presented on the lower internal graphic) which group the graphic's pixels depending on the class of the object the pixel is representing. However, this could not be enough for the image similarity assessment task. In some cases, the decision-maker may also want to assess image similarity while considering the quantity of objects belonging to different classes. Therefore, for the development of this algorithm, the panoptic semantic image segmentation algorithm has been used [11]. It differs from the standard segmentation model in the fact, that it can distinguish different objects belonging to the same class.

Of course, the success of image segmentation and detecting objects that it contains strongly relies on the accuracy of the chosen segmentation algorithm. It is worth mentioning that each semantic segmentation algorithm can recognize a limited set of object classes C and deliver information about objects that can be transformed into a chosen set of features F. The main reason for choosing semantic segmentation as a tool in feature extraction task is that its output is provided as a set of image pixels subsets representing instances of detected classes. This form of data is explicit and more human-understandable, and what comes of it, it is also easier to analyze or work with. Tools used in corresponding stages in similar solutions almost always use convolutional neural networks for feature extraction tasks. This approach produces a single vector representation of an overall image. This embedding implicitly stores information about objects and their features, which are hard to operate in the context of a single object presented in it.

The second stage of the proposed approach is tasked with transforming the outcome from the feature extraction stage into a computer-understandable form. The data obtained from the previous step is modified to the form of a matrix V with N rows (number of classes which can be recognized by the chosen panoptic segmentation algorithm – cardinality of set C) and K columns (number of features extracted for each object in the previous step of the algorithm – cardinality of set F) like it was presented in Table 1. Each row existing within this data structure denotes all detected objects belonging to an c_n ($n \in \langle 1; N \rangle$) class, that can be recognized by the feature extraction step. Whereas each column of the mentioned matrix denotes distinguishable feature f_k ($k \in \langle 1; K \rangle$) of the object belonging to class c_n. For example, the columns defined in a V matrix can have the following meaning: vertical and horizontal position of objects belonging to a given class, quantity of objects, vertical and horizontal size, etc. Value v_{nk} at the intersection of row and column in the matrix presented in Table 1 describes how much an object belonging to class c_n reproduces a feature f_k. Each of these measures is de facto a membership function from fuzzy sets theory [12]. Therefore, the value of v_{nk} may vary from 0 to 1. Usage of this type of measure simplifies feature processing and ensures, that particular feature values will not have a disproportionate impact on the final similarity estimate, due to their magnitude. Additionally, for every image passed to the algorithm a set U is created. It contains unique classes of objects detected in the input image.

Tab. 1. An example of image segmentation output transformed into matrix

	f_0	f_1	…	f_k
c_0	v_{00}	v_{01}	…	v_{0k}
c_1	v_{10}	v_{11}	…	v_{1k}
…	…	…	…	…
c_n	v_{n0}	v_{n1}	…	v_{nk}

The last stage of the proposed approach is tasked with determining the similarity of the image embeddings produced by the second step of the solution. So, a method for comparing spatial representations of graphics is needed. Standard systems that perform image comparisons, such as CBIR systems, almost always use vectors with implicitly stored information about objects in the image and the provided graphics. Working with this data structure is not flexible. For example, a decision-maker is not capable of putting a stronger emphasis on some specific feature or the presence of an object of a specific class. Due to this fact the last stage of the proposed approach will introduce a method of comparing image matrix representations, which is more flexible and susceptible to applying the decision-maker preferences.

Assessing image similarity will consist of a few steps ending with providing the decision-maker a single fuzzy value $s \in [0,1]$. If this value is close to unity, it means that images provided to the algorithm are similar in the context desired by the decision-maker. But if the similarity measure is close to 0, it means that the provided images are entirely different.

The process of determining images likeness starts with defining feature weight matrix W, class importance vector I and penalty factor p by the decision-maker. First of those factors has a size of N rows and K columns. Each of the values defined in this matrix $w_{nk} \in [0,1]$ $(n \in N, k \in K)$ will describing how important feature n of objects representing class k is for the decision--maker. All grades in the feature weight matrix will be by default set to value of one. With the class importance vector I the decision-maker will be capable of specifying, which classes of objects should have a greater impact on final images likeness value. Each value of this data structure i_n $(n \in \langle 1; N \rangle)$ will describe how important for the decision-maker the objects representing class c_n are in context of image similarity assessment process. Of course, like in previous data structures, each of the class importance vector values will be set to value of one by default. The last decision-maker's defined value – penalty factor p – describes how much situation in which the given class c_n is present only in one image affects the final value of image similarity measure. By default, this factor value will be set to 0,5.

In the next step, the algorithm will subtract corresponding values v_{nk} from both the base and the compared image embedding matrices. The absolute values from the previous operation are then subtracted from one. This is done to invert fuzzy values to bring them closer to unity for similar images. Each result of the subtraction is then multiplied by the corresponding feature weight value w_n. Next, the outcoming matrix is summed by rows, creating a vector containing a single value per the model class. Every value of this vector will be multiplied by one if objects representing corresponding class c_n are observed in both images. If a class c_n objects are only present in one of the images, then the corresponding vector value is multiplied by the value of one less penalty factor. Otherwise, it is multiplied by zero. Next, each of the values from the mentioned data structure is multiplied by the class importance factor i_n and divided by the number of distinguishable features K. After completing the previous calculations, the result is summed up by column. Finally, the outcome is divided by the sum of sets intersection cardinality and the difference of images' unique classes union cardinality and intersection cardinality multiplied by the penalty factor p. The resulting value s_1 is a measure of image similarity, where the intersection of detected class sets has at least one element. If its value is close to unity, it means that the graphics provided to the algorithm are very similar in terms of the context and the decision-maker's preferences. Otherwise, the graphics are different. The mathematical formula (1) represents all mathematical operations performed during the assessing similarity of two images, that share at least one detected object of the same class. There is also the possibility that the two images supplied to the algorithm will not have objects belonging to the same classes. In this case, the similarity value will be predefined and will be 0. Both cases are included in formula (2).

$$
\begin{aligned}
s_1 = \\
= \frac{1}{\overline{\overline{U_1 \cap U_2}} + \left(\overline{\overline{U_1 \cup U_2}} - \overline{\overline{U_1 \cap U_2}}\right) * p} * \\
* \sum_{n=0}^{N} \frac{i_n}{K} * ([c_n \in (U_1 \cap U_2)] + (1 - p) * \\
* [c_n \notin (U_1 \cap U_2) \wedge c_n \in (U_1 \cup U_2)]) * \\
* \sum_{k=0}^{K} (1 - |v_{1nk} - v_{2nk}|) * w_{nk}
\end{aligned}
\tag{1}
$$

$$s = \begin{cases} 0 & \overline{\overline{U_1 \cap U_2}} = 0 \\ s_1 & \overline{\overline{U_1 \cap U_2}} > 0 \end{cases} \quad (2)$$

4. Experiments

The architecture presented in Image 1 and described in this article's previous chapter was implemented using the Mask2Former panoptic image semantic segmentation model with the Swin backbone. It was tested on the COCO 2017 validation images dataset consisting of 5 thousand graphics depicting various real-world objects. Initially, all the images were processed by the panoptic semantic segmentation algorithm and the results of this operation were preserved for future operations.

This section of the article will discuss the results of two conducted experiments. Tests were carried out with each of the values from the feature weight matrix W and class importance vector I set to 1. This was done to inform the algorithm that every object's features and object class are equally important in the process of assessing image similarity.

The first experiment involved selecting three randomly chosen reference images from the COCO 2017 validation pictures and comparing them with two other photos. These other compared graphics were selected from the set of images for each reference image, following a specific criterion: the first image closely resembles the reference image, while the second image differs in terms of the context and contained objects. The results of this test are depicted in Figure 3.

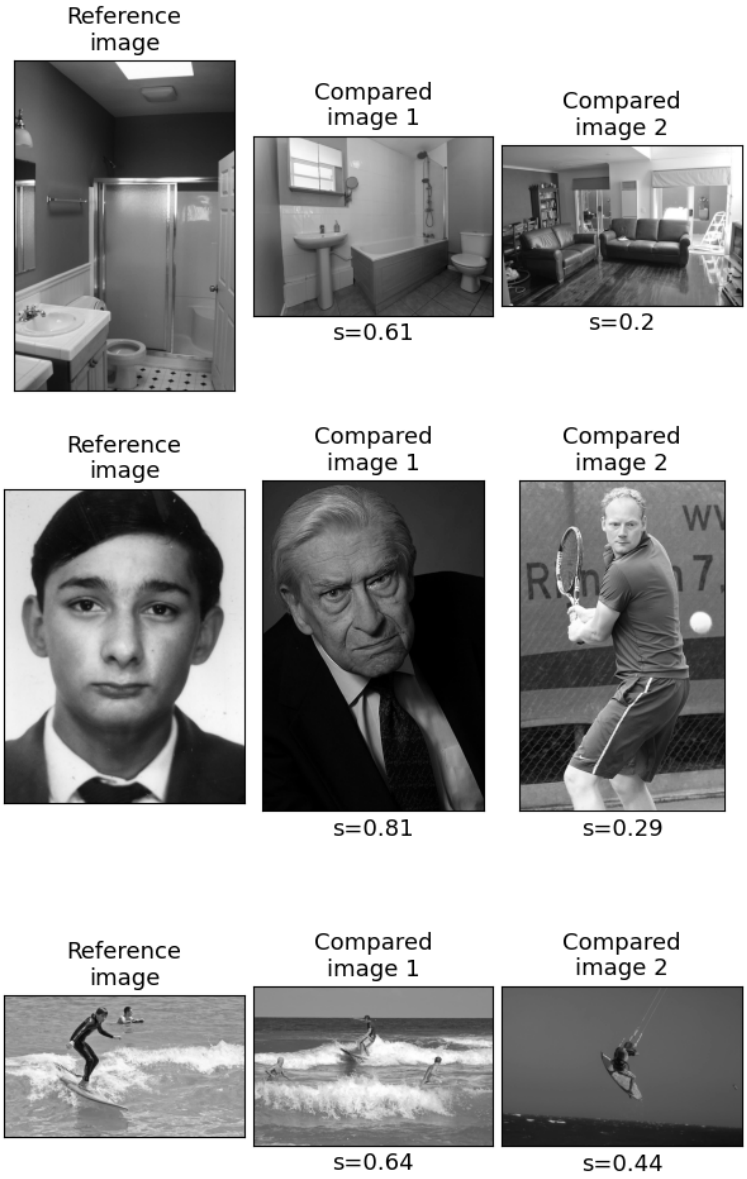

Fig. 3. The results of the first test

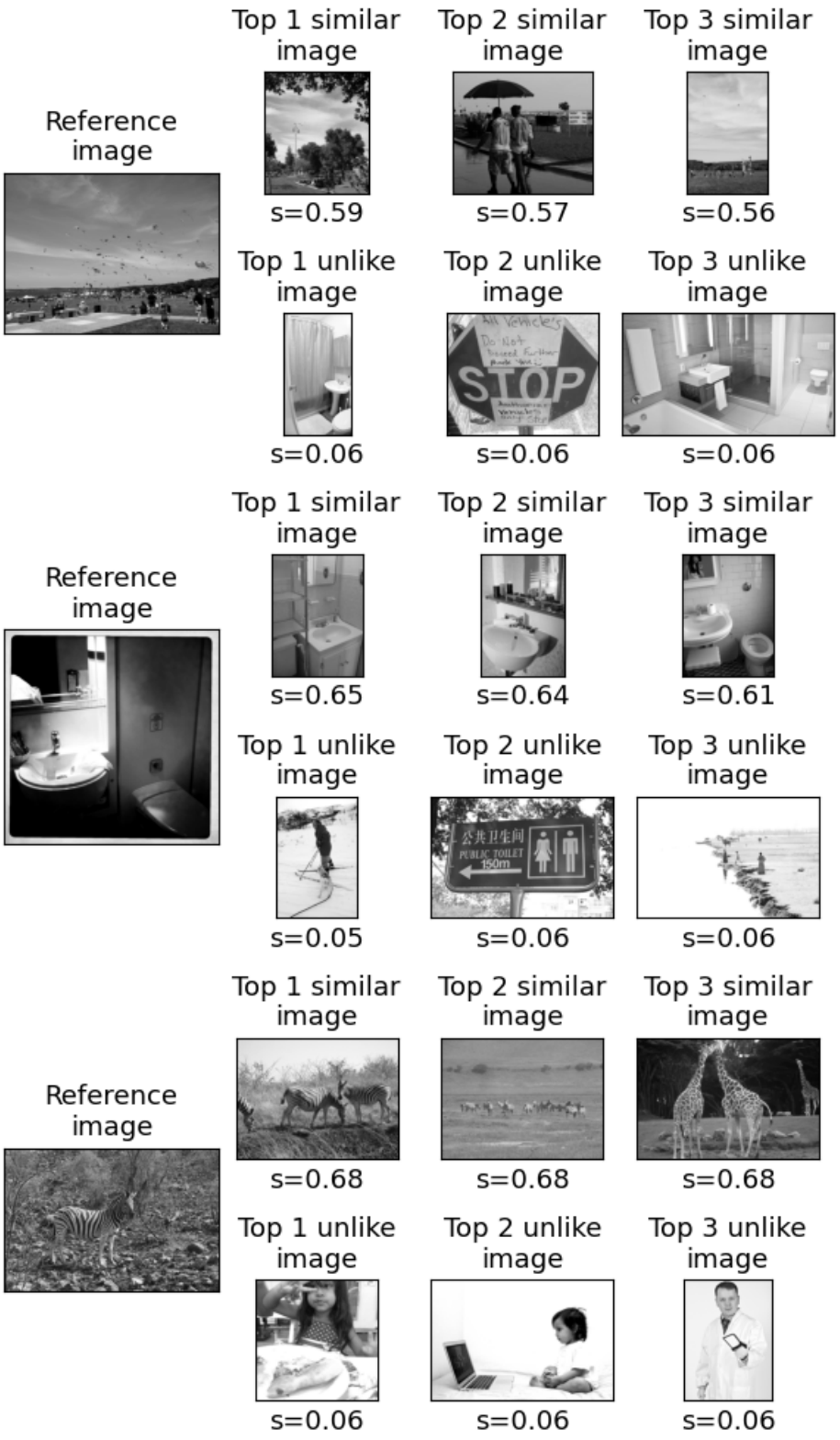

Fig. 4. The results of the second test

As can be seen, the algorithm assessed the compared images' similarity score as expected. Pictures in the second column of Figure 3 received higher similarity scores than graphics from the third column. This is because the images in the first and second columns contain almost the same sets of depicted objects. Even though the images in the third column look similar, they received a lower similarity measure score. This is because the pictures either show different sets of objects (as in the first and second-row images) or the same objects arranged differently (as in the third-row image).

The second test tasked the algorithm with selecting three images with the highest similarity score and three photos with the lowest likeness value. Alike pictures were sought in the entire COCO 2017 validation set. The results of this test are depicted in Figure 4.

As seen in Figure 4, the algorithm has identified both the most similar and the most dissimilar images. Three pictures from the first inner rows are noticeably like the reference photo. This is because graphics presented to the algorithm contained a subset of reference image's objects. On the other hand, the images presented in the second inner rows are completely different. They received the lowest score because they did not share any objects with the reference image.

5. Conclusions

This article presents a novel approach for evaluating image similarity using information acquired from images through semantic segmentation algorithms. It also allows the decision-maker to specify, which classes of objects or which object's features should have a greater impact on the final image similarity score. The conducted tests have demonstrated that the proposed solution is suitable for tasks such as comparing images and searching for similar graphics within sets. Hence, the proposed solution could be used in the future to build a CBIR (Content-Based Image Retrieval) system. It is worth emphasizing that the effectiveness of the proposed solution largely depends on the quality of the selected semantic segmentation algorithm. This is because these models are not universal. They can only identify specific classes of objects that they have been trained on.

6. Bibliography

[1] Shahzeb H., Prayas D., Shaayan H., "Image Processing in Artificial Intelligence", *International Journal of Scientific Research in Computer Science*, *Engineering and Information Technology*, Vol. 6, No. 5, 244–249 (2020).

[2] Wang X., Kitani K.M., Hebert M., „Contextual Visual Similarity", 2016, https://doi.org/10.48550/arXiv.1612.02534.

[3] Dubey S.R., "A Decade Survey of Content Based Image Retrieval Using Deep Learning", *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 32, No. 5, 2687–2704 (2022).

[4] Zheng W., Ouyang Y., Ford J., Makedon F., "Ontology-based Image Retrieval", *Ijetrm Journal*, Department of Computer Science, Dartmouth College, USA, https://doi.org/10.5281/zenodo.3352813.

[5] Schroff F., Kalenichenko D., Philbin J., „FaceNet: A Unified Embedding for Face Recognition and Clustering", *2015 IEEE Conference on Computer Vision and Pattern Recognition* (*CVPR*), p. 815–823, Boston, MA, USA, 2015.

[6] Bandikolla P., Reddy V.R.K., "Image retrieval using a combination of keywords and image features", Master Thesis, School of Engineering Blekinge Institute of Technology, Ronneby, Sweden, 2008.

[7] Bromley J., et al., "Signature Verification using a 'Siamese' Time Delay Neural Network", *Advances in Pattern Recognition Systems Using Neural Network Technologies*, series in: Machine Perception and Artificial Intelligence, Vol. 7, p. 25–44, Singapore 1994.

[8] Redmon J., Divvala S., Girshick R., Farhadi A., "You Only Look Once: Unified, Real-Time Object Detection", 2016, https://doi.org/10.48550/arXiv.1506.02640.

[9] Xu H., Ma Y., Liu H., Deb D., Liu H., Tang J., Jain A.K., "Adversarial Attacks and Defenses in Images, Graphs and Text: A Review", 2019, https://doi.org/10.48550/arXiv.1909.08072.

[10] Khan M.A., Dhotre D., "CBIR: A review on its new trends in current era", *International Research Journal of Modernization in Engineering Technology and Science*, Vol. 5, No. 6, (2023).

[11] Kirillov A., He K., Girshick R., Rother C., Dollár P., "Panoptic Segmentation", 2019 *IEEE/CVF Conference on Computer Vision and Pattern Recognition* (*CVPR*), p. 9396–9405, Long Beach, CA, USA, 2019.

[12] Zadeh L.A., "Fuzzy sets", *Information and Control*, Vol. 8, No. 3, 338–353 (1965).

Oparty na treści pomiar podobieństwa obrazów bazujący na informacjach pozyskanych z algorytmów segmentacji semantycznej

R. WASILUK

Celem niniejszego artykułu jest zaprezentowanie nowatorskiego sposobu zapisywania informacji zawartych na obrazach w ustrukturyzowanej formie oraz przeprowadzania procesu szacowania podobieństwa obrazów z użyciem wspomnianych struktur danych. Rozwiązanie zaprezentowane w tym dokumencie opiera swoje działanie na analizie wyników otrzymanych od wstępnie wytrenowanych algorytmów segmentacji semantycznej. Rezultaty te mogą zostać przetransformowane do postaci zbioru wektorów, których wartości będą reprezentowały cechy obiektów wykrytych na dostarczonych obrazach. Takie struktury danych mogą zawierać istotne informacje na temat detekcji algorytmu np.: położenie wykrytego obiektu na obrazie, rozmiar wykrytego obiektu w porównaniu do wielkości całej grafiki, kolor dominujący itp. Przygotowane w taki sposób wektorowe reprezentacje obrazów mogą być porównywane między sobą przy użyciu wielu narzędzi matematycznych takich jak miary odległości. Co więcej zaprezentowane w niniejszym artykule podejście pozwala decydentowi zdefiniować wartość wagi każdej z cech dla poszczególnych klas obiektów. Pozwala to modelować preferencje decyzyjne oraz sprawia, że podzbiór cech obiektów może mieć większy wpływ na ostateczną wartość podobieństwa obrazów od pozostałych parametrów.

Słowa kluczowe: podobieństwo obrazów, segmentacja semantyczna, wektorowa reprezentacja obrazów.