Detection of Similarity Between Images Based on Contrastive Language-Image Pre-Training Neural Network

Vasyl Lytvyn[1] , Roman Peleshchak[1], Ihor Rishnyak[1], Bohdan Kopach[1] and Yuriy Gal[2]

[1] *Lviv Polytechnic National University, 12 Stepana Bandery Street, Lviv, 79000, Ukraine*[1]
[2] *Drohobych Ivan Franko State Pedagogical University, Drohobych, 24 Ivan Franko St., 82100, Ukraine*

Abstract

The process of evaluating image similarity is a complex task, made more challenging by the complexity of the objects of study. As computational power continues to advance, it is becoming increasingly clear that neural networks are taking center stage in addressing a wide array of computer vision challenges. This study introduces a novel approach to this problem by leveraging the capabilities of the CLIP model. The unique feature of the proposed solution is that the calculation of similarity uses not the vector representation of the image, but the vector representation of textual descriptions, which were selected and encoded by the CLIP model. During the experimental stage, an image encoder based on ResNet-50 and a text encoder based on the BERT neural network were used. The results of this research are promising, showing that the proposed method outperforms traditional methods like SSIM and FSIM by demonstrating higher accuracy and robustness in capturing perceptual image similarities. This indicates that the CLIP model is a suitable tool for comparing complex images that feature a multitude of objects and layers. The methodology presented in this work holds potential for a variety of applications where image comparison plays a crucial role, such as in semantic image search, image categorization, and clustering. This approach opens up new avenues for exploring image similarity, offering a fresh perspective that combines the visual and textual domains, utilizing CLIP model encoders.

Keywords

Artificial neural networks, CLIP, computer vision, image similarity detection

1. Introduction

With the growth of graphical data, the task of finding connections and relationships between images has become an important task that may find practical applications in solving various computer vision tasks. Identifying connections between images helps in mapping out how different images relate to each other within a collection. This is important for organizing the data effectively, making it easier to navigate through large datasets, and understanding the overarching themes or categories present. It can uncover hidden patterns, such as the repetition of specific objects or themes, which might not be immediately apparent, thereby aiding in understanding the complexity and diversity of modern datasets.

By finding connections between images, systems can better interpret search queries to return more relevant results. This involves analyzing the images' visual and contextual similarities, allowing the return of more nuanced results that go beyond mere categorization by tags. Furthermore, this approach facilitates the creation of hierarchical graphs of image collections, which might significantly enhance the organization and accessibility of large datasets.

This hierarchical graphing enables the visualization of data at various levels of granularity, from broad categorizations down to finely detailed relationships, providing a multi-layered

COLINS-2024: 8th International Conference on Computational Linguistics and Intelligent Systems, April 12–13, 2024, Lviv, Ukraine

vasyl.v.lytvyn@lpnu.ua (V. Lytvyn); roman.m.peleshchak@lpnu.ua (R. Peleshchak); ihor.v.rishnyak@lpnu.ua (I. Rishnyak); bohdan.v.kopach@lpnu.ua (B. Kopach); yuriyhal@gmail.com (Y. Hal)

0000-0002-9676-0180 (V. Lytvyn); 0000-0002-0536-3252 (R. Peleshchak); 0000-0001-5727-3438 (I. Rishnyak); 0009-0002-5158-589X (B. Kopach); 0009-0006-1771-532X (Y. Hal)

understanding of the dataset's structure. Such a structured representation is invaluable for tasks requiring detailed analysis of the connections and similarities within the dataset, such as advanced image retrieval systems, recommendation engines, and even the training of more sophisticated machine learning models that can learn from the complexity of relationships rather than just the presence of similar features.

Neural networks can be an essential mechanism for finding the connections between images. The active development of models for image recognition has only accelerated the development of tools for solving this task. A particularly important milestone was the introduction of the CLIP model in 2021, which allows establishing relationships between text and images [1]. The flexibility of the model enables it to be adapted to a wide variety of tasks.

The aim of this work is to create a mechanism that, using the CLIP model, can automatically analyze images and determine a numerical measure of image similarity. The proposed image similarity scoring mechanism utilizes CLIP model image and text encoders to seamlessly bridge the gap between visual and textual data, enabling a comprehensive analysis that incorporates both the semantic content of images and the contextual nuances of associated text. By leveraging the sophisticated capabilities of the CLIP model, this method can identify relationships between images by finding the best matching descriptions, transforming them into the same vector space, and comparing them through cosine similarity. During the experiments, the BERT [2] model was used for encoding text. Its application as a text encoder is due to its universality, accuracy, ease of training, and the ability to compare the semantic similarity of text through vector representation of sentences. For the encoding of images, the ResNet-50 [3] model was selected due to its robust performance in deep learning tasks related to image recognition. This model is distinguished by its deep convolutional neural network architecture, which incorporates residual connections to facilitate the training of deeper networks by alleviating the vanishing gradient problem.

The flexibility of the proposed solution enables precise adjustments and customization to meet the unique needs of different domains. This adaptability guarantees that the proposed mechanism can be seamlessly incorporated into a variety of systems and platforms that stand to gain from identifying relationships between images.

2. Related works

The concept of CLIP revolves around training a model using a vast collection of images paired with corresponding textual descriptions [1]. This approach enables the model to grasp and generalize visual concepts in a manner that resonates with human understanding. At its core, CLIP comprises two primary elements: an image encoder and a text encoder. The former processes and transforms input images into feature vectors, while the latter does the same for textual descriptions. The training goal is to enhance the similarity between feature vectors of matching pairs of images and text, while reducing the similarity for non-matching pairs, employing a contrastive learning strategy. A notable strength of CLIP is its capability for zero-shot learning, allowing it to adapt to new tasks post-training without the need for further fine-tuning. This makes it applicable for tasks like image classification and object detection, where it can operate based on relevant textual descriptions. The language-driven design of CLIP facilitates a more adaptable and user-friendly interaction with the model. By enabling users to direct the model's actions through simple natural language prompts, it enhances the potential for collaboration between humans and machines. Additionally, this approach lowers the barrier to entry, making the model more approachable for individuals lacking extensive technical knowledge. In our work, we explored the potential of this model to quantify the similarity between relationships.

The task of finding similarities between images extends naturally into the construction of image graphs and ontologies, representing a more structured and interconnected approach to understanding visual data. Image graphs are visual representations where nodes correspond to individual images and edges represent the relationships or similarities between them. In work [4] the researchers describe a graph-based methodology for the analytical examination of extensive collections of images and texts. Through the analysis of a given image corpus, they ascertain the degrees of similarity among images and the semantic distances among texts, thereby constructing a composite graph representation. A significant limitation of this approach is the prerequisite for accurate annotations of all images involved. This requirement underscores the importance of precise metadata or annotations in leveraging graph-based techniques for effective visual analytics and relationship mapping in image and text datasets.

Ontologies in the context of image analysis serve as a framework for organizing and categorizing images based on a hierarchy of concepts or classes. By defining a set of relationships and properties within a domain, ontologies help in structuring data in a way that reflects real-world relationships. They add a layer of semantic depth, enabling the classification and retrieval of images based on a comprehensive understanding of their content, relationships and attributes. In the study [5] an innovative ontological bagging approach is introduced, which leverages discriminative weak attributes across multiple learning instances. This method employs the bagging technique to reduce error propagation across classifiers. The research utilizes an ensemble consisting of VGG-16[6], ResNet-50 and Xception[7] models to extract a comprehensive feature set. These features are then utilized by classifiers, which are trained within an ontological framework, to execute the image classification task. This approach enhances the accuracy and reliability of classifying forest images by effectively combining deep learning models with ontological insights. However, constructing a comprehensive ontology to uncover relationships within an image collection presents several challenges.

Clustering and finding relationships between images are complementary techniques in the realm of data analysis and computer vision, each with its unique approach to understanding and organizing visual information. At their core, both methods rely on the extraction and analysis of features from images—such as color, texture, shape, or deep learning embeddings—to discern patterns and similarities within large datasets. The study [8] introduces a clustering method that utilizes a shared nearest neighbors strategy, applicable to both content-based features and textual tags. This method's underlying principle posits that objects sharing a higher number of common neighbors are more likely to belong to the same cluster. It calculates weighted connections between objects based on shared neighbors and categorizes each object into one of three classes: core, noise, or aggregate, thereby enhancing the precision and utility of clustering in managing tagged image collections.

The study [9] explores the application of various clustering techniques to uncover relationships within and organize large datasets. This research showcases how different clustering algorithms can be effectively utilized to segment big collections of images, facilitating their analysis, visualization, and tagging. While finding connections involves identifying and mapping relationships between individual items based on specific scoring criteria or attributes, clustering groups data points into subsets or clusters based on similarity measures without necessarily mapping the intricate relationships between each point within or across clusters.

The methodology presented in [10] incorporates the use of a semantic information extraction tool alongside a visual layout creation mechanism. The extraction tool leverages a convolutional neural network image captioning technique to generate descriptive captions for images, which are then converted into semantic keywords. Concurrently, the layout creation mechanism utilizes an innovative co-embedding model that aligns images and their corresponding semantic keywords within the same two-dimensional space. A significant benefit of this approach is its capacity for automated image annotation and the conversion of word embeddings into vector space. However, a notable limitation arises from the reliance on word-level embeddings, which may not always sufficiently encapsulate the full breadth of an image's content. This discrepancy highlights the potential need for more nuanced or comprehensive methods to fully convey the complexity and richness of visual data, a gap that the CLIP model is well-suited to bridge. Unlike methods dependent solely on word-level embeddings, CLIP can work with text embeddings containing multiple words, allowing for a deeper understanding of both textual and visual information.

3. Method

To train the CLIP model, pairs of text and images are used, which are fed into encoders that transform the data into vector representations. This process allows the model to learn from a wide range of visual and textual information, facilitating the understanding of complex concepts across different domains. Given pairs of image descriptions (terms) and images, the transformation process into vectors can be represented by the following formula:

$$\overline{v_{I_i}} = f_I(I_i) \tag{1}$$

$$\overline{v_{T_i}} = f_T(T_i), \tag{2}$$

where i is the ordinal number of the pair, $i \in \{1, 2, \dots, n\}$, n – is the number of pairs;
$\overline{v_{I_i}}$ – vector representation of the image;
$\overline{v_{T_i}}$ – vector representation of the textual description;
f_I – image encoder;
f_T – text encoder;

The task of training the CLIP model is to optimize the parameters of the similarity function S, which can be used to find correspondences between image-text pairs:

$$sim(\overline{v_{I_i}}, \overline{v_{T_i}}) = S(\overline{v_{I_i}}, \overline{v_{T_i}}) \tag{3}$$

This function aims to maximize the similarity between representations of corresponding images and texts, thus facilitating effective cross-modal understanding. Using the function S for any image, we can find N descriptions with the highest similarity score (sim). Utilizing the text encoder f_T based on BERT architecture, it becomes feasible to compute the pairwise cosine distances between the textual descriptions corresponding to two images, denoted as I and I'.

$$D = \frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{N}\frac{\overline{v_{T_i}}\,\overline{v_{T'_k}}}{\left\|\overline{v_{T_i}}\right\|\left\|\overline{v_{T'_k}}\right\|}, \tag{4}$$

where i is the index of the textual description of image I;
k – the index of the textual description of image I';
$\overline{v_T}$ – the vector representation of the textual description of image I;
$\overline{v_{T'}}$–the vector representation of the textual description of image I';

The calculation of pairwise cosine similarity between vector representations of textual descriptions, which CLIP considers the best matches for pairs of images, enables the determination of the similarity level between these images. This approach is founded on the assumption that images with similar content will be associated with similar textual descriptions by CLIP. Consequently, their vector representations in the shared image-text space will be closer to each other. This metric quantifies the similarity between images based on how CLIP interprets their content through associated text. Instead of a subjective assessment of the similarity of two images, we obtain a quantitative indicator that can be compared across different experiments or even between different models.

Using CLIP to calculate similarity between images in practice can be highly effective due to its ability to compare images even without knowing their initial descriptions. CLIP can effectively measure their similarity based on the semantic understanding it has gained during pre-training, thereby facilitating robust image analysis and retrieval tasks. This capability is particularly advantageous in scenarios where manual annotation or description of images is impractical or unavailable, allowing for efficient and scalable image processing pipelines. The number of descriptions used to calculate image similarity is an important parameter that must be carefully chosen. If the number is too high, there is a risk that the cosine distance between vectors created by the encoder may not accurately represent the true semantic similarity between images. This is because overly complex or redundant descriptions can introduce noise and dilute the meaningful information captured by the embeddings. On the other hand, if the number of descriptions is too low, the model may not capture enough contextual information to accurately assess similarity.

Given that for any pair of images $D > 0$, it becomes practical to incorporate a scalar parameter τ (threshold) to build relationship graph between images. This parameter becomes essential to prevent the formation of correlations, or groupings, between semantically distinct images. The introduction of τ facilitates the clustering of similar images, thereby ensuring the avoidance of erroneous associations and improving the accuracy of the similarity measurement process.

4. Experiments

For pre-training ResNet-50 image encoder, we utilized model pretrained on the ImageNet-1k dataset [11], which contains around one million photographs belonging to 1000 different categories. For training the CLIP model, we used the public datasets Flickr8k [12] and its extension Flickr30k [13]. These datasets are commonly applied for training models specialized in image recognition or description. Together, the datasets comprise nearly 40000 illustrations, each accompanied by 5 unique descriptions that can vary in length and content, providing a diverse range of textual data for model training. The images were carefully selected to ensure that they are suitable for use in a machine learning context without infringing on any individual's rights. Additionally, the datasets used, such as ImageNet and Flickr, have guidelines and policies in place to address privacy and ethical concerns.

The evaluation process was conducted on images with semantically and structurally diverse characteristics to ensure that the model's performance was tested across a wide range of scenarios. This approach helped to ascertain the model's robustness and adaptability in handling various types of visual information, thereby providing a comprehensive understanding of its capabilities. By including images that varied in terms of content, style, and complexity, the evaluation aimed to mimic real-world conditions where the model might be deployed. This rigorous testing methodology not only highlighted the strengths of the model in accurately identifying and interpreting diverse visual cues but also exposed any potential limitations or areas for improvement. The training process involved a sequence of standard transformations applied to each image. The images were first resized to a dimension of 256x256 pixels, followed by cropping the central region to a dimension of 224x224 pixels to emphasize the most significant part of the image. Normalization was performed on each of the RGB channels to ensure that the pixel values were in a similar range, which is crucial for the effective training of neural networks. This preprocessing step is common in image classification tasks as it helps in reducing computational complexity while retaining the essential features of the images.

To transform textual descriptions, we utilized a BERT tokenizer with an output vector size capped at 256, enabling the conversion of text into a format compatible with the BERT model's processing capabilities. This preprocessing pipeline ensures that the input data is consistent and optimized for the learning algorithms, facilitating efficient and effective model training. Since we employ BERT as the text encoder, utilizing this model to assess the accuracy of CLIP ensures methodological consistency in our experiments. BERT's architecture, based on transformers, is capable of detecting complex semantic connections in textual data. This guarantees that during the evaluation, the effect of text encoder structure on model accuracy is considered.

The model training was conducted utilizing the PyTorch framework [14], with parameter optimization achieved through the Adam optimizer [15] at a learning rate of 1e-5. This approach was adopted to ensure incremental updates of the model's weights. A batch size of 5 was maintained, and the total image count was capped at 35,000 to achieve a balance between computational efficiency and the need for a diverse and representative dataset.

For each of the five training epochs of the CLIP model, the identical images were utilized, yet they were shuffled randomly along with their corresponding text descriptions to inject variation in the sequence of presentation. This strategy aimed to enhance the training process by exposing the model to diverse arrangements of the same data set in each epoch. By doing so, the model is encouraged to learn more robust and generalized features, reducing the risk of overfitting to a specific order of data.

The cross-entropy loss function was used for training. The utilization of the cross-entropy loss function was crucial in assessing the model's ability to accurately associate text with images, thereby enhancing the performance of the encoders. The CLIP model, designed to interpret both visual and textual data, heavily relies on the loss function for effective integration of these two data types. The selection of an appropriate loss function is essential for the model's performance, as it ensures stable training and guarantees that semantically similar images or text fragments are positioned closer in vector space.

5. Results

The effectiveness of the CLIP model in identifying relationships between images was evaluated by comparing its results with established image similarity metrics, such as SSIM [16] and FSIM [16]. This comparison aimed to assess the model's ability to detect semantic and structural similarities in images, which are critical aspects measured by both metrics. By analyzing how the CLIP model's performance aligns with these metrics, the study sought to determine if the CLIP model could offer a deeper understanding of image relationships beyond what traditional similarity measures can provide.

To assess the effectiveness of the proposed solution in measuring image similarity, we calculated similarity scores for each pair of images in the evaluation dataset. For each metric, we employed different similarity thresholds, meaning that if the value was below the specified threshold, the images were considered dissimilar. Subsequently, we determined the number of correctly identified image pairs. The results are presented in Table 1.

Table 1
Image Similarity Calculation Results

Metric	Threshhold(τ)	Accuracy
SSIM	0.7	64%
FSIM	0.3	56%
CLIP-score	0.5	**82%**

Compared to the standard model, the approach based on the CLIP model didn't demonstrate significant gains in finding relationships between images in the training dataset. For example, in Figure 1, we can see that the scores are quite similar, and all three compared methods correctly identified the relationships between images.

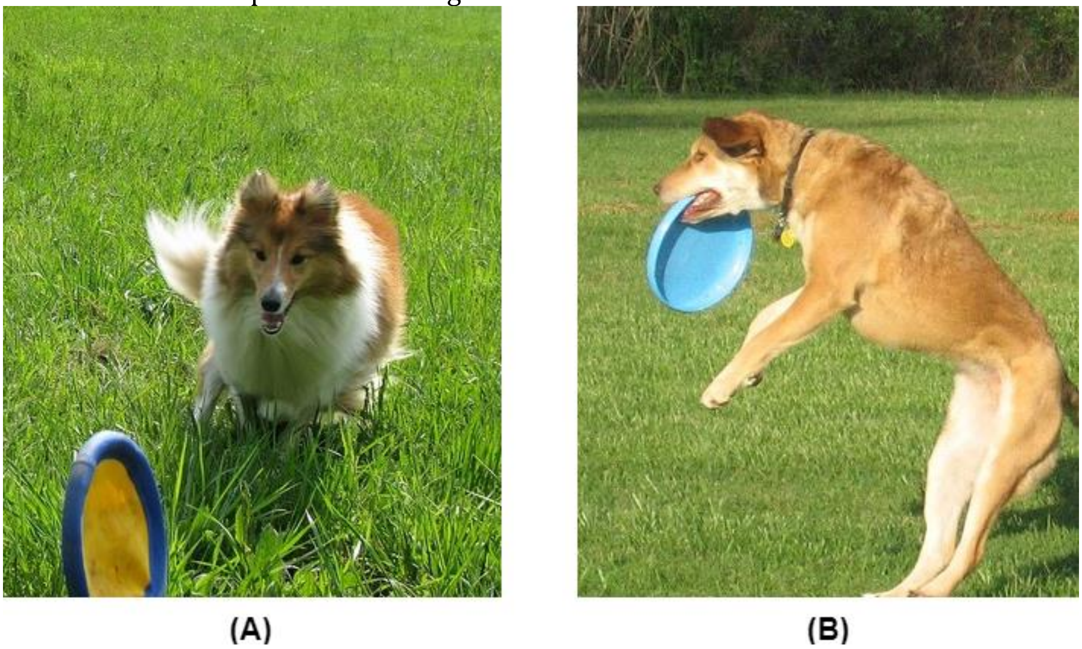

Figure 1: Comparison of Image A and Image B. The similarity scores are as follows: FSIM = 0.34, SSIM = 0.81, CLIP-score = 0.65.

Nevertheless, it showed promising results in terms of generalization and robustness when applied to a diverse set of images in the test dataset. Figures 2, 3 and 4 show that CLIP can accurately identify images that are perceptually similar or have differences, even when FSIM and SSIM metrics indicate errors and their scores fall below (or above) the threshold outlined in the Table 1. The CLIP-based image similarity score model demonstrates its superiority in identifying semantic relationships, such as distinguishing between a "sitting dog" and simply a "dog," or discerning subtle differences between "running" and "walking" poses in human figures. This ability to recognize subtle differences and understand the context within images sets it apart from traditional models that might rely solely on pixel-level comparisons.

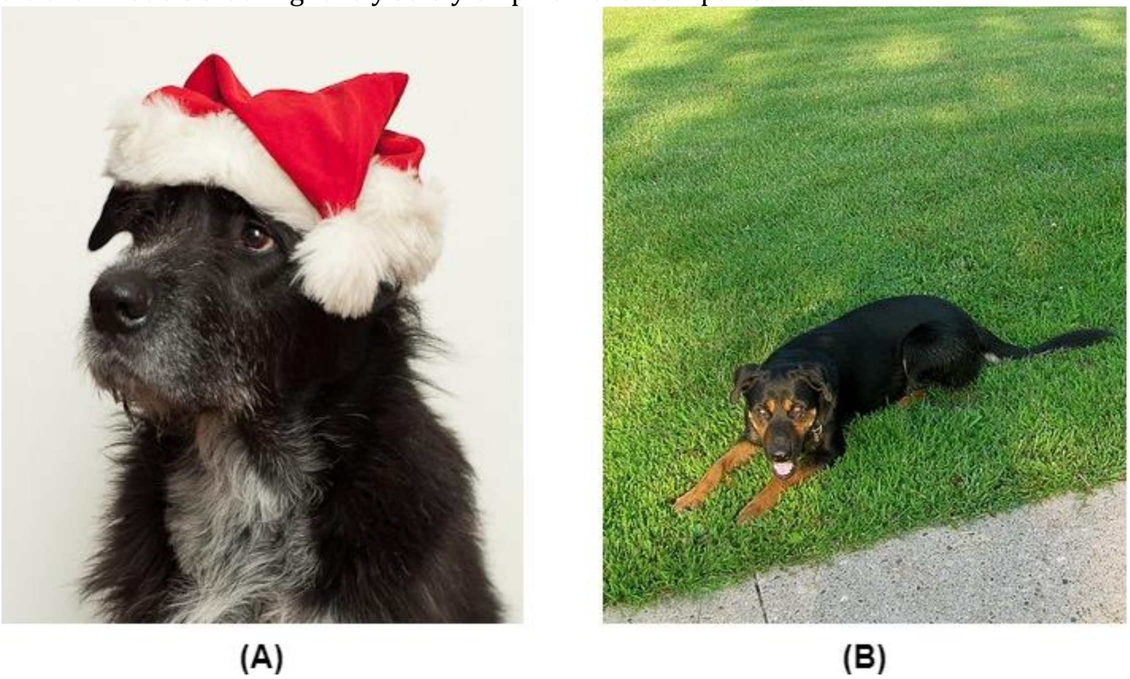

Figure 2: Comparison of Image A and Image B. The similarity scores are as follows: FSIM = 0.28, SSIM = 0.69, CLIP-score = 0.67.

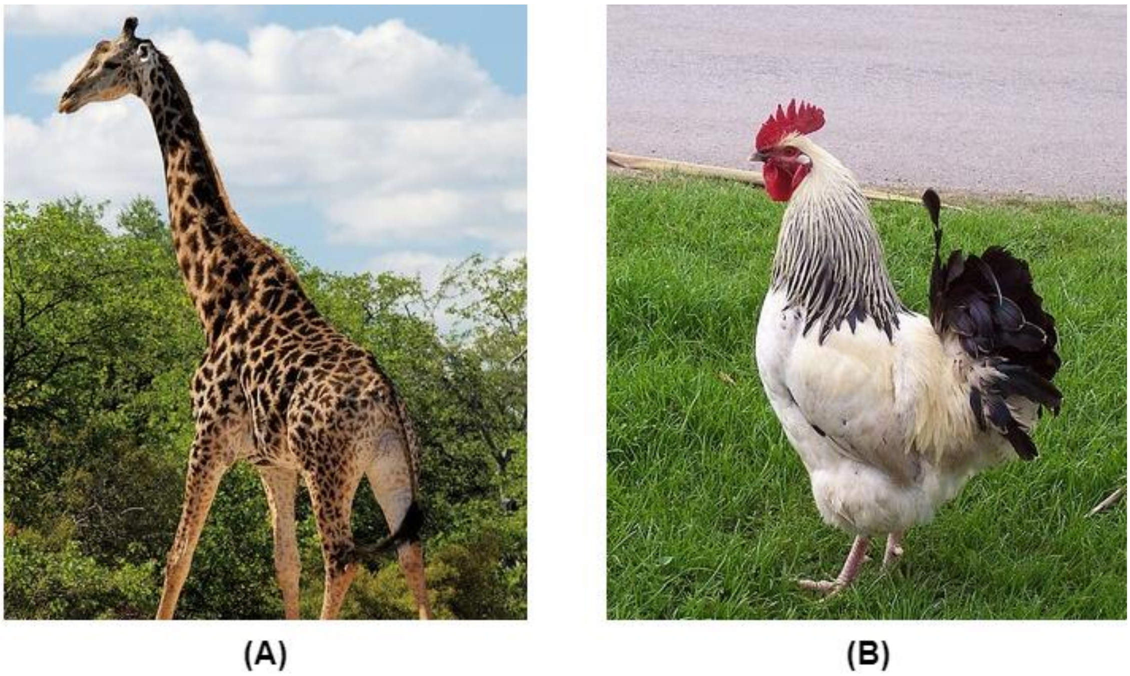

Figure 3: Comparison of Image A and Image B. The similarity scores are as follows: FSIM = 0.34, SSIM = 0.80, CLIP-score = 0.21.

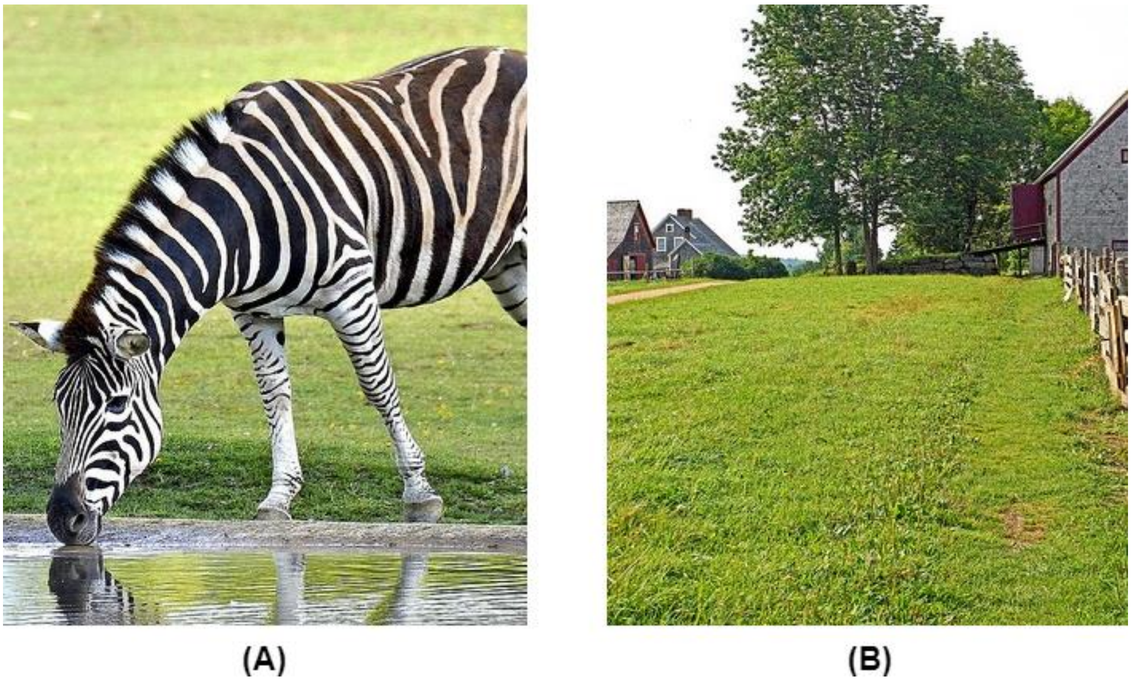

Figure 4: Comparison of Image A and Image B. The similarity scores are as follows: FSIM = 0.32, SSIM = 0.72, CLIP-score = 0.28.

By harnessing the rich semantic information encoded in its embeddings and leveraging the power of the BERT text encoder, the CLIP-based approach provides a more nuanced and comprehensive understanding of image content compared to traditional pixel-based methods , which often rely solely on the visual features of an image. This often leads to a superficial interpretation of the image content, focusing primarily on the appearance rather than the underlying context or semantics. This integration of visual and linguistic features enables the model to capture the subtleties of image details.

6. Discussion

The training process of the CLIP model was challenging due to its dual nature, which involves handling both image and text data simultaneously. This duality requires the model to learn complex visual-linguistic representations, making the training process more intricate than that of models focused on a single modality. Additionally, ensuring that the image and text components of the model are effectively aligned and integrated adds another layer of complexity to the training process. During training, it is essential to focus on the quality of textual data, as generating inaccurate text vectors can result in less than optimal model performance. Incorrect or noisy text data can produce misleading representations, which may hinder the model's ability to accurately link images with their corresponding textual descriptions. This can result in poor performance in identifying relationships between images.

The results indicate that the image comparison method based on the CLIP model is effective and outperforms methods such as FSIM and SSIM. This approach allows for the analysis of images and the scoring of their similarity, making it applicable in various tasks such as image retrieval, categorization, and visual search. Despite its effectiveness, there are definitely areas for improvement.

In experiments, the model showed excellent performance in comparing images similar to those in the training dataset. However, when evaluating images not well-represented in the training data, the results were often less reliable. Therefore, it is advisable to pre-train the model on a diverse and comprehensive dataset to improve its generalization capabilities. Moreover, it is essential to continuously update and refine the model with new data to ensure its effectiveness across various real-world scenarios. If there was not a single text description among the set that can describe the image, the result might be unsatisfactory, which means that it is crucial to have a rich and varied set of textual descriptions to effectively describe and interpret images. This highlights the importance of not only having a diverse set of images in the training dataset but also ensuring that the textual descriptions are comprehensive and cover a wide range of possible scenarios and characteristics.

The text encoder, which identifies textual descriptions for an image, is crucial in the developed method. The vectors it produces are used for comparison with image vectors and with vector representations of the text of another image to assess image similarity. Therefore, the encoder's ability to understand semantic relationships between sentences is important, as deficiencies can negatively affect the outcome. During experiments, we repeatedly noticed that although BERT typically finds semantic connections between sentences, the cosine similarity between vectors of sentences with similar structure but describing completely different objects is usually higher than that of sentences with different structures but similar objects. This suggests that the model may prioritize structural similarities over semantic content in some cases, which could lead to misinterpretations of the actual meaning and relevance of the sentences. This observation indicates a potential area for improvement. Addressing this issue could involve refining the model to better balance the importance of structural or semantic features, or experimenting with different language models that can be used as text encoders, thereby enhancing its ability to accurately interpret and compare sentences based on their true semantic content, detecting tonality of the sentence [17, 18].

A significant drawback of the proposed approach, identified during the model evaluation process, is that calculating the similarity between images requires identifying the most appropriate textual descriptions from all available descriptions. This process can be time-consuming, as enhancing results often involves searching for multiple optimal textual descriptions for comparison with those of other images. To mitigate this issue, several strategies can be employed, such as refining the search algorithm for more efficient navigation through the extensive set of descriptions, adopting caching methods or storing precomputed text vectors to expedite the retrieval process. Furthermore, leveraging parallel processing or distributed computing can decrease the time needed to find the best textual descriptions, rendering the approach more feasible for real-world applications.

7. Conclusions

In the research, the approach to using cosine similarity for evaluating the accuracy of similarity between images is leveraged, utilizing the capabilities of the CLIP model. This method utilizes trained encoders for both text and images, allowing it to effectively evaluate the semantic similarity between images. By analyzing the content and context of the images, the metric provides a more nuanced understanding of their similarity, beyond just pixel-level comparisons. The results showcase the metric's ability to accurately match similar images, demonstrating a high capacity to understand complex images containing multiple objects. This performance surpasses that of traditional metrics such as FSIM and SSIM, highlighting its effectiveness in capturing the nuanced details and semantic relationships within images.

The proposed image comparison metric offers a versatile tool for addressing a wide range of computer vision challenges. It can be effectively employed in semantic search applications, where it can enhance the accuracy of retrieving relevant images based on their content. In image classification tasks, the metric can contribute to more precise categorization by understanding the semantic similarities between different images. Additionally, it holds significant potential for recommendation systems, where it can be used to suggest visually similar products to users, thereby improving the user experience and increasing engagement.

Although the metric demonstrates satisfactory results, it still faces limitations related to performance, a high dependency on the data used for training the model, and accuracy in analyzing complex images with multiple objects. These limitations underscore the need for further refinement and testing, especially in real-world scenarios where computational efficiency and robustness are crucial.

Future work should explore addressing these issues, optimizing the metric, and adapting it to a broader range of tasks. This could involve experimenting with different algorithms and architectures to reduce the metric's reliance on training data and improve its generalizability. Investigating the impact of using different types of text encoders on the model's accuracy is particularly important. Utilizing a model that positions semantically related sentences closer in vector space will inherently enhance the metric's accuracy. This could lead to more precise image comparisons and better performance in tasks such as image retrieval and classification. Additionally, examining how alterations in the image encoder influence the metric's parameters is also crucial. Changes in the image encoder could affect the metric's sensitivity to visual features and its ability to capture semantic similarities.

References

[1] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, et al., Learning transferable visual models from natural language supervision, in: Proceedings of the International Conference on Machine Learning, ICML '21, PMLR, 2021, pp. 8748–8763.

[2] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of deep bidirectional transformers for language understanding, in: J. Burstein, C. Doran, T. Solorio (Eds.), Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), Association for Computational Linguistics, Minneapolis, Minnesota, 2019, pp. 4171–4186. doi:10.18653/v1/N19-1423.

[3] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 770–778. doi:10.1109/CVPR.2016.90.

[4] Y. Gu, C. Wang, J. Ma, R. J. Nemiroff, D. L. Kao, iGraph: a graph-based technique for visual analytics of image and text collections, in: Proceedings of the Visualization and Data Analysis 2015, VDA '15, SPIE, 2015, vol. 9397, pp. 53–67. doi:10.1117/12.2074198.

[5] C. Kwenda, M. Gwetu, J. V. Fonou-Dombeu, Ontology with deep learning for forest image classification, Applied Sciences 13 (8) (2023) 5060.

[6] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, in: Proceedings of the 3rd International Conference on Learning Representations, ICLR 2015, Conference Track Proceedings, San Diego, CA, USA, May 7-9, 2015. URL: http://arxiv.org/abs/1409.1556.

[7] F. Chollet, Xception: Deep learning with depthwise separable convolutions, in: Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 1800–1807. doi:10.1109/CVPR.2017.195.

[8] P.-A. Moëllic, J.-E. Haugeard, G. Pitel, Image clustering based on a shared nearest neighbors approach for tagged collections, in: Proceedings of the 2008 International Conference on Content-Based Image and Video Retrieval, CIVR '08, Association for Computing Machinery, New York, NY, USA, 2008, pp. 269–278. doi:10.1145/1386352.1386390.

[9] K. Pogorelov, M. Riegler, P. Halvorsen, C. Griwodz, ClusterTag: Interactive Visualization, Clustering and Tagging Tool for Big Image Collections, in: Proceedings of the 2017 ACM on International Conference on Multimedia Retrieval, ICMR '17, Association for Computing Machinery, New York, NY, USA, 2017, pp. 112–116. doi:10.1145/3078971.3079018.

[10] X. Xie, X. Cai, J. Zhou, N. Cao, Y. Wu, A Semantic-Based Method for Visualizing Large Image Collections, IEEE Transactions on Visualization and Computer Graphics 25 (7) (2019) 2362–2377. doi:10.1109/TVCG.2018.2835485.

[11] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, L. Fei-Fei, ImageNet Large Scale Visual Recognition Challenge, International Journal of Computer Vision 115 (3) (2015) 211–252. doi:10.1007/s11263-015-0816-y.

[12] M. Hodosh, P. Young, J. Hockenmaier, Framing image description as a ranking task: Data, models and evaluation metrics, Journal of Artificial Intelligence Research 47 (2013) 853–899. doi:10.1613/jair.3994.

[13] P. Young, A. Lai, M. Hodosh, J. Hockenmaier, From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions, Trans. Assoc. Comput. Linguistics 2 (2014) 67–78. doi:10.1162/TACL_A_00166.

[14] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, et al., PyTorch: An imperative style, high-performance deep learning library, Advances in Neural Information Processing Systems 32 (2019).

[15] D. P. Kingma, J. Ba, Adam: A Method for Stochastic Optimization, in: Y. Bengio, Y. LeCun (Eds.), 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, 2015. URL: http://arxiv.org/abs/1412.6980.

[16] U. Sara, M. Akter, M. Uddin, Image Quality Assessment through FSIM, SSIM, MSE and PSNR—A Comparative Study, Journal of Computer and Communications 7 (2019) 8–18. doi:10.4236/jcc.2019.73002.

[17] R. Peleshchak, V. Lytvyn, I. Peleshchak, A. Khudyy, Z. Rybchak, S. Mushasta, Text Tonality Classification Using a Hybrid Convolutional Neural Network with Parallel and Sequential Connections Between Layers, in: Proceedings of the 6th International Conference on Computational Linguistics and Intelligent Systems (COLINS 2022), Volume I: Main Conference, Gliwice, Poland, May 12-13, 2022, CEUR Workshop Proceedings, vol. 3171, CEUR-WS.org, 2022, pp. 904–915. URL: https://ceur-ws.org/Vol-3171/paper65.pdf.

[18] Z. Zhang, Y. Wu, H. Zhao, Z. Li, S. Zhang, X. Zhou, X. Zhou, Semantics-Aware BERT for Language Understanding, in: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, no. 05, 2020, pp. 9628–9635. doi:10.1609/aaai.v34i05.6510.