Exploring Human Perception-Aligned Perceptual Hashing

Jelle De Geest
IDLab, Ghent University - imec

Patrick De Smet and Lucio Bonetto
National Institute of Criminalistics and Criminology (NICC)

Peter Lambert, Glenn Van Wallendael, and Hannes Mareen
IDLab, Ghent University - imec

***Abstract*—The widespread sharing of images has led to challenges in controlling the spread of harmful content in consumer devices, particularly Child Sexual Abuse Material (CSAM). Perceptual hashing offers a solution by enabling the fast detection of blacklisted images through compact representations of visual content. However, automatically detecting (near-)duplicate images in overwhelming volumes of data is challenging due to the limitations of traditional perceptual hashing methods. For example, existing methods can fail to detect images with minor modifications, specifically spatial modifications. Additionally, they were often designed to find images derived from the same original image, and hence are incapable of recognizing visually similar images that originate from a different acquisition origin. This study explores the use of Vision Transformers (ViTs), specifically the CLIP model, to enhance perceptual hashing, better aligning with human perception. The proposed ViTHash method is compared against traditional perceptual hashing methods like pHash, dHash, and PDQHash. Quantitative results show that ViTHash outperforms traditional methods in handling spatial distortions such as rotation and mirroring, although it is less robust to visual quality distortions like blurring and compression. Qualitative analysis reveals that ViTHash aligns more closely with human perception, and is capable of identifying visually similar images, even when they are images depicting visually similar content yet originate from different acquisition origins. These findings demonstrate that ViTHash offers significant potential for applications requiring nuanced image similarity assessments, providing a valuable tool to enhance the detection of illicit content in consumer electronics devices, and support law enforcement efforts. The source code can de downloaded at https://github.com/JelleDeGeest/ViTHash.**

IN TODAY'S DIGITAL ERA, the widespread sharing of images and multimedia has fundamentally changed the way we communicate and consume information.

Digital Object Identifier 10.1109/MCE.YYYY.Doi Number

Date of publication DD MM YYYY; date of current version DD MM YYYY

Published by the IEEE Consumer Technology Society

The internet and digital devices have made it easier than ever to connect, share, and explore. However, with these benefits come significant challenges, especially in controlling the spread of harmful or sensitive content. A critical concern here is the alarming rise in Child Sexual Abuse Material (CSAM), with reported files to the National Center for Missing & Exploited Children (NCMEC) increasing from 450,000 in the early 2000s to an astounding 105 million by 2023 [1].

Law enforcement agencies face an overwhelming volume of data, making it extremely challenging to manually sift through vast collections of media to identify illicit content. The emotional toll on investigators, coupled with the time-consuming nature of this task, highlights the need for (semi-)automated approaches. By automating the process of searching through and identifying CSAM, we can significantly reduce the time required to take action, minimize prolonged suffering of victims, and protect investigators from the severe psychological trauma associated with manually reviewing such disturbing media. Law enforcement agencies maintain extensive databases of known illicit content, and being able to efficiently search these databases for known content within large volumes of data is essential.

In consumer electronics, efficiently comparing images is essential for content moderation, copyright protection, and imagesearch [2], [3]. Social media platforms, cloud storage services, and photo management applications should be able to automatically detect duplicates or visually similar images, even when they have been altered. For example, platforms should identify and remove inappropriate content, while photo libraries can group similar images to make it easier for users to organize and manage large collections. These examples highlight the importance of (near-)duplicate image detection in everyday consumer technology experiences.

To address these challenges, a hierarchical workflow of hashing techniques is employed: starting with binary hashing to detect exact duplicates, followed by perceptual hashing to identify visually similar images despite minor modifications. Perceptual hashing generates a kind of digital ”DNA” for images, allowing media content to be compared for similarities. Using this technique, one can automatically scan large data volumes to find blacklisted images. It is important that the algorithms are robust against manipulations that do not significantly alter the perceptual content, and as such find all images that originate from the same source image.

(a) Image A (b) Image B

Figure 1: Example of a visually similar image pair from a different acquisition origin. This is different from the typical perceptual hashing evaluation use case where we examine visually similar (distorted) images originating from the same source image.

While traditional perceptual hashing methods have been useful tools, they still fall short in some important ways [4], [5], [6]. That is, although they are typically robust against visual quality distortions (such as compression, noise, and blurring), they typically lack robustness against spatial operations such as mirroring or cropping (as demonstrated in the evaluation section of this paper). Moreover, small adversarial alterations to an image can make it unrecognizable by these methods [7]. Furthermore, they often fail to align well with human perception of similarity. For example, it may be beneficial to find images that do not originate from the same source image, yet are visually similar (e.g., images from the same scene taken moments apart, as in **Figure 1**) hereafter referred to as visually similar images with a different acquisition origin. In the evaluation, it is shown that traditional perceptual hashing methods fail to detect such visually similar images with different acquisition origin. By enhancing perceptual hashing methods to more accurately detect (spatially) modified and visually similar content, new and circulating material can be identified more effectively, ultimately supporting a faster and more reliable intervention.

This study explores using pretrained Vision Transformers (ViTs) to enhance perceptual hashing, with the goal of improving robustness and alignment with human visual perception. The proposed method is called ViTHash, and was briefly presented in previous work [8]. The effectiveness of ViTHash is compared to traditional perceptual hashing methods. The aim is to

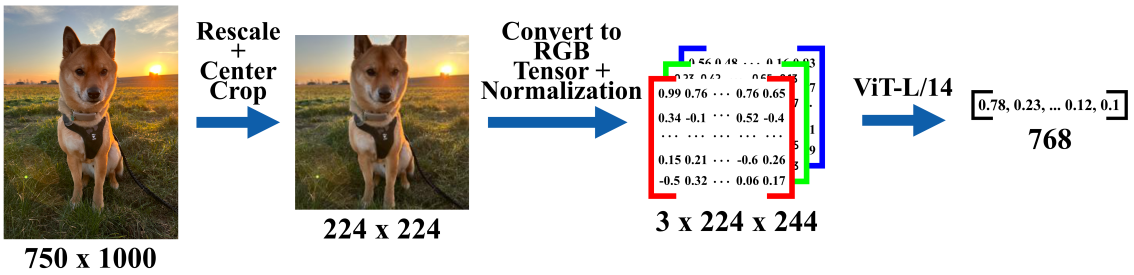


Figure 2: Overview of the proposed ViTHash method, going from an input image to a perceptual hash, utilizing a pretrained ViT-L/14 model.

get a better view on the value that large pretrained AI-based methods can bring in the automated detection of visually similar images. Compared to previous work [8], this paper provides (1) more background context, (2) a more extensive evaluation (using a larger dataset of 100,000 images instead of 2,000 images in previous work, and using the more robust 256-bit variants of the traditional perceptual hashing methods instead of the 64-bit versions), and (3) presents new experimental results on the detection of visually similar images with a different acquisition origin.

The advancement of AI-based perceptual hashing methods, such as the proposed ViTHash, is increasingly relevant to the consumer electronics industry. The ability to quickly detect manipulated or harmful images is essential in smart devices and cloud storage systems. Consumer technologies may include automated image analysis for content moderation. Improving these methods helps devices more effectively filter inappropriate content and stop its spread, making digital environments safer.

The remainder of this paper is organized as follows. First, existing perceptual hashing methods are outlined, and background about ViTs is given. Then, the proposed ViTHash method is discussed in-depth, of which an overview is displayed in **Figure 2**. Subsequently, ViTHash is evaluated by comparing its performance to other perceptual hashing methods, as well as by utilizing ViTHash for detecting visually similar images that do not originate from the same source. Next, we contextualize the proposed method with related work, and discuss potential future work directions that address ViTHash's limitations. Finally, the work is concluded.

TRADITIONAL PERCEPTUAL HASHING

Traditional perceptual hashing methods have significantly advanced the field of digital forensics by enabling more nuanced image comparison techniques that closely mirror human visual perception. This section begins by introducing the concept of perceptual hashes and outlining their key properties. Subsequently, several commonly used perceptual hashing algorithms (pHash, dHash, and PDQHash) are explored.

Perceptual Hashing

Perceptual hashing is designed to generate hash values that reflect the visual content of images. Unlike binary hash functions [9], which is the first step to detect exact duplicates, perceptual hashes aim to produce similar hashes for images that are visually alike. This characteristic makes perceptual hashing particularly valuable for applications [10], [11], [12] such as image deduplication, copyright enforcement, and the detection of illicit content like Child Sexual Abuse Material (CSAM).

The effectiveness of perceptual hashing hinges on several key properties. The most important property is similarity sensitivity; minor alterations to an image should result in only slight changes to its hash, allowing for the identification of visually similar images despite modifications such as resizing, cropping, or color adjustments. Efficiency is also critical, as the hashing and retrieval process needs to be swift to handle large volumes of data. Additionally, perceptual hashes must resist reversibility to prevent the original image from being reconstructed from its hash, thereby safeguarding sensitive content. Lastly, while collision resistance is less stringent than in cryptographic hashing, it remains important that different images do not produce identical hashes, thereby minimizing false positives in similarity detection.

Traditional Methods

Several traditional perceptual hashing algorithms have been developed, each with different design strategies and strengths. Among the most notable are pHash [13], [14], dHash [15], [14], and PDQHash [16], which are discussed in this subsection. These three methods are widely recognized in the literature, frequently serve as baselines for comparison, and represent distinct approaches. Their prevalence in both academic research and industry applications makes them well-suited for illustrating the capabilities and limitations of traditional perceptual hashing. An overview of the discussed methods and their key features is displayed in Table 1.

pHash [13], [14], short for perceptual hash, leverages frequency analysis to identify key components of an image's visual structure. By analyzing the (low-frequency) DCT coefficients of an image, pHash focuses on the overall patterns and textures that define its appearance. The reliance on low-frequency components makes pHash effective in capturing the fundamental visual characteristics that are less susceptible to superficial changes.

PDQHash [16], developed by Meta, builds upon the principles of pHash by incorporating advanced filtering techniques to enhance robustness against small image alterations. PDQHash introduces a preprocessing step that applies blurring filters to the image before generating the hash. This additional step ensures that the hash is more resilient to subtle changes and noise, improving its ability to consistently identify visually similar images despite various modifications.

dHash [15], [14], or difference hash, adopts a different strategy by concentrating on the relative differences in pixel brightness across an image. Instead of analyzing frequency components, dHash examines how pixel values change from one region to another. By assessing the gradients and transitions in brightness, dHash produces hash values that reflect the structural and textural variations within the image.

While perceptual hashing has significantly advanced digital forensics by enabling image comparisons that better align with human visual perception than binary hashing, these methods have notable limitations. They are susceptible to adversarial attacks where minor pixel alterations can preserve visual similarity but cause substantial differences in hash values, allowing images to evade detection [7]. Additionally, traditional perceptual hashes often fail to align with human perception, resulting in irrelevant or incorrect matches when searching large datasets. This is demonstrated in the evaluation. For example, these methods are highly sensitive to common spatial modifications such as mirroring, rotation, and cropping. This can be partly overcome by calculating multiple hashes for different orientations, at the cost of increasing database size and retrieval times. However, it is infeasible to exhaustively calculate hashes for all potential spatial transformations.

Moreover, traditional methods are unable to handle visually similar images with a different acquisition origin, as they are designed to match images based solely on identical or near-identical source content.

These shortcomings highlight the need for more robust perceptual hashing techniques that can better withstand adversarial changes, closely mimic human visual similarity assessments, and detect content-level similarities across images, even when they originate from different moments or perspectives.

PRETRAINED VISION TRANSFORMERS

The advent of Vision Transformers (ViTs) [17] has significantly advanced the field of computer vision [18], offering an alternative to traditional convolutional neural networks (CNNs). Whereas the Transformer architecture was originally designed for natural language processing [19], ViTs apply this concept to images by treating them as a series of patches. A prominent application of ViTs is in OpenAI's Contrastive Language-Image Pre-training (CLIP) model, which aligns visual information with textual descriptions in a shared embedding space. This section provides more background information on ViTs and CLIP.

Vision Transformer Architecture

The Vision Transformer adapts the Transformer architecture for image data by splitting an image into a sequence of fixed-size patches, similar to tokens in text processing. Each patch is linearly embedded, combined with positional embeddings to retain spatial information, and then processed through Transformer encoder layers. The self-attention mechanism within the Transformer allows the model to capture both local and global dependencies across the image.

Specifically, the ViT-L/14 model divides an input image of size 224×224 pixels into 14×14 patches, resulting in 256 patches. Each patch is flattened and projected into a high-dimensional embedding vector.

Table 1: Overview of Different Perceptual Hashing Methods

Method	Approach	Key Features
pHash	DCT-based frequency analysis	Emphasizes low-frequency components to capture overall patterns, robust against slight changes in brightness or color
dHash	Pixel difference hashing	Focuses on local brightness gradients for structural comparisons, extremely simple and fast
PDQHash	DCT-based frequency analysis + filtering	Similar to pHash but more robust to subtle changes due to the filtering step
ViTHash	Vision Transformer embeddings	Captures nuanced visual similarity by aligning with human perception, excels in detecting spatial and semantic content changes

These embeddings are then fed into multiple layers of the Transformer encoder, as illustrated in **Figure 3**.

CLIP Overview

The CLIP model [17] is designed to learn visual concepts from natural language supervision by embedding images and text into a shared representation space. It consists of two separate encoders: a Vision Transformer (e.g., ViT-L/14) that processes images, and a masked self-attention Transformer that processes text descriptions. The text encoder uses a Transformer-based architecture, where masked self-attention mechanisms allow the model to focus on relevant words in the text while understanding their contextual relationships. This helps the model create embeddings that capture the meaning and nuances of the text.

During training, CLIP employs a contrastive learning approach using a large dataset of image-text pairs. The objective is to bring the embeddings of matching image-text pairs closer together while pushing non-matching pairs further apart in the shared embedding space. This is achieved using a contrastive loss function, which calculates the similarity between all possible pairs in a training batch. The loss function encourages higher similarity for true image-text pairs and lower similarity for mismatched pairs, thereby learning a representation where related visual and textual information are closely aligned.

Figure 4 illustrates the overall architecture of the CLIP model, showcasing both the image and text encoders and their alignment through contrastive learning.

PROPOSED METHOD: ViTHash

Building upon the strengths of the CLIP model's pretrained Vision Transformer, ViTHash is proposed as a perceptual hashing method designed to enhance image similarity detection. An overview of how the method transforms an image into a hash is given in Figure 2.

ViTHash generates perceptual hashes by extracting image embeddings from the pre-trained CLIP model (ViT-L/14). The hash generation process involves two main steps. First, input images are rescaled and center cropped to 224×224 pixels to match the CLIP model's expected input size. Second, the preprocessed image is passed through the ViT-L/14 encoder to obtain a 768-dimensional embedding vector. Since each element in the embedding is represented as a 16-bit floating-point number, the resulting perceptual hash size is 1,536 bytes.

Similarity Measurement Using Cosine Similarity

To compare perceptual hashes from ViTHash, the concept of cosine similarity is leveraged, which measures the alignment between embedding vectors in a high-dimensional space. As already indicated above, when the pre-trained CLIP model (ViT-L/14) processes an image, it outputs a 768-dimensional embedding vector, with each element represented as a 16-bit floating-point number. Cosine similarity between two embedding vectors A and B is calculated as:

$$\text{Cosine Similarity}(A, B) = \frac{A \cdot B}{\|A\| \, \|B\|} \qquad (1)$$

A higher cosine similarity indicates a stronger alignment between the embeddings, suggesting that the images are more similar in content. By directly using this similarity metric, ViTHash takes advantage of the semantic understanding encoded in the embeddings to identify modified or related content effectively.

ViTHash is exploratory, and future work could focus on optimizing the size of the perceptual hash. One approach could be training a new head that takes the embedding space as input and is specifically optimized for this use case, potentially reducing the hash size while maintaining performance.

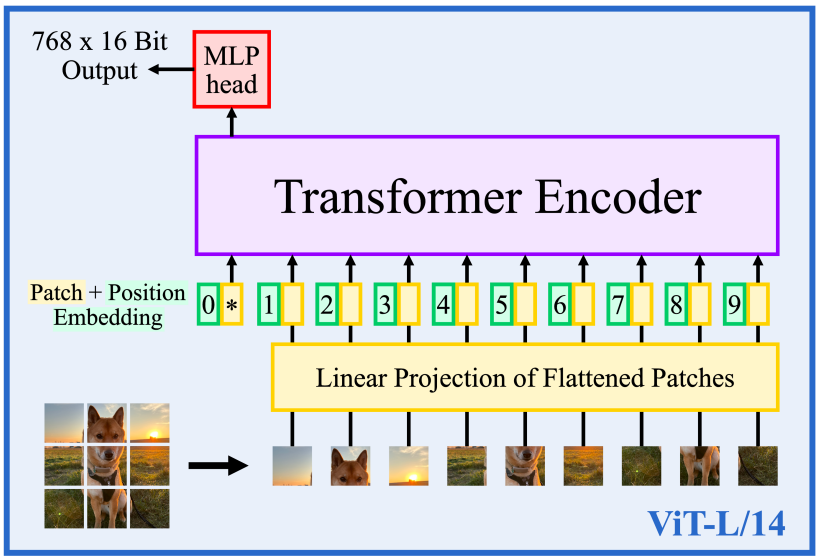


Figure 3: Vision Transformer architecture used for image encoding, showing the sequence of operations from input patches to output embeddings.

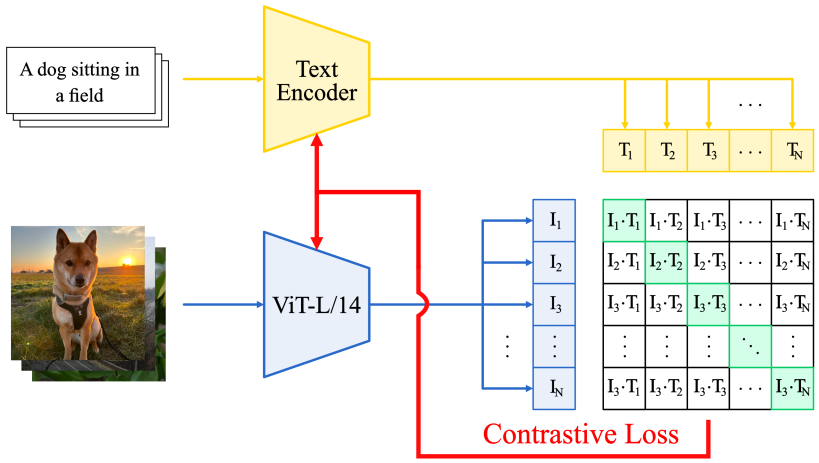


Figure 4: Architecture of the CLIP model [17] , comprising the ViT-L/14 Vision Transformer (shown in Figure 3) for image encoding and the masked self-attention Transformer for text encoding. The model aligns image and text embeddings in a shared feature space using contrastive learning to facilitate tasks such as image similarity detection and zero-shot classification.

EVALUATION

This section presents the evaluation of the proposed ViTHash method in comparison with traditional perceptual hashing techniques such as pHash, dHash, and PDQHash.

First, images are matched with distorted versions of themselves, in order to assess how effectively each method can detect those modified versions. The evaluation begins by outlining the experimental setup, which involves simulating realistic conditions through various image distortions and transformations. Following this, quantitative results are provided that include computational performance metrics (hash generation time, hash size, and comparison time) and robustness metrics (F1-scores) across different types of image distortions. Qualitative results are then offered to gain deeper insights into the strengths and limitations of ViTHash and traditional methods, examining how they align with human perception.

The final subsection explores scenarios where images are visually similar but do *not* have the same acquisition origin (e.g., images from the same scene

taken moments apart, as shown in Figure 1). This examination aims to assess each method's ability to detect visual similarity beyond identical or distorted versions of the same source image, highlighting ViTHash's potential in capturing human-like perceptions of similarity in a broader context.

Experimental Setup

To assess the performance of ViTHash, an experimental setup was developed that simulates realistic conditions where image modifications are frequent. All experiments were carried out on a desktop computer featuring an Intel Core i9-9900K CPU and an NVIDIA RTX 2080 SUPER GPU.

For this study, the DISC21 dataset [20] was utilized, which was originally created by Meta for the Image Similarity Challenge at NeurIPS'21. From this dataset, a subset of 50,000 images was extracted to build a database of hashes using four different hashing techniques: pHash, dHash, PDQHash, and ViTHash. In previous work [8], results for the traditional methods were presented for 64-bit hashes. In this paper, results are reported for the larger 256-bit variants, as they offer more accurate results than smaller sizes. Moreover, PDQHash strongly recommends using at least 256 bits for reliable performance. It should be noted that the ViTHash hash size is significantly larger (1,536 bytes), as it retains the full 768-dimensional embedding vector with 16-bit floating-point representation for each element.

To simulate realistic adversarial attacks and evaluate the robustness of ViTHash, 50,000 images were selected from the database along with an additional 50,000 images from a separate partition of the dataset, ensuring no overlap between the sets. These 100,000 images were then subjected to a range of transformations, namely rotation (90 Degrees), scaling (50%), blurring (Gaussian with radius 2), mirroring, color distortion (50%), compression (JPEG with quality factor 50), and crop rotation (15%). These transformations were used to create modified versions of the images that emulate potential real-world attacks.

Quantitative Results

This subsection first evaluates the computational performance of each hashing method using two metrics: hash generation time and hash size. Then, the robustness against image distortions is evaluated using the F1 score.

Hash generation time reflects the computational cost to produce a hash, while hash size indicates the memory needed to store it. While these metrics provide a useful indication of performance, the absolute values depend on factors like batch size and hardware, meaning results may vary across different setups.

The quantitative results are presented in **Table 2**. In terms of hash generation time, PDQHash is the slowest, whereas pHash and dHash are the fastest. In terms of hash size, ViTHash is significantly larger than the others (1,536 B vs. 32 B).

Table 2: Hash generation time and hash size for different hashing methods

Hashing Method	Generation	Size
pHash	62.3 ms	32 B
dHash	66.8 ms	32 B
PDQHash	512.3 ms	32 B
ViTHash	213.5 ms	1536 B

The F1 score is the harmonic mean of precision and recall, providing a single metric that balances both concerns. It is particularly useful when there is a need to balance the trade-off between precision and recall. The F1 score is calculated as:

$$F1\ Score = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} \quad (2)$$

A high F1 score indicates a good balance between precision and recall, making it a robust measure in scenarios where both false positives and false negatives are critical. Unlike the accuracy score, the F1 score accounts for the possibility of a single match producing multiple false positives. Therefore, it serves as the primary performance metric in this evaluation.

Table 3 presents the F1 score results for the various hashing methods across different image distortions. The distortions can be categorized into two groups: spatial distortions (rotation, mirroring, and crop rotation) and visual quality distortions (blurring, color distortion, compression, and rescaling).

ViTHash shows a notable strength in handling spatial distortions, outperforming the traditional methods, which struggle with recognizing images modified in this way. Conversely, for visual quality distortions, the classical methods provide excellent performance, whereas ViTHash shows a lower F1 score, indicating some sensitivity to alterations in visual quality. These results are in line with the previous research [8] comparing ViTHash and pHash, which utilized the AUC of ROC curves on a smaller sample size.

Table 3: F1 score comparison for different perceptual hashing methods across various image modifications

Distortion	F1 Score			
	pHash	dHash	PDQHash	ViTHash
Blur	**0.9999**	0.9842	**0.9999**	0.8868
Color Distortion	**0.9999**	0.9713	**0.9999**	0.9666
Compression	0.9998	0.9807	**0.9999**	0.9431
Rescaling	**0.9998**	0.9808	0.9997	0.9185
Crop Rotation	0.0009	0.0001	0.0002	**0.5766**
Mirroring	0.0000	0.0003	0.0001	**0.8039**
Rotation	0.0000	0.0000	0.0001	**0.3478**

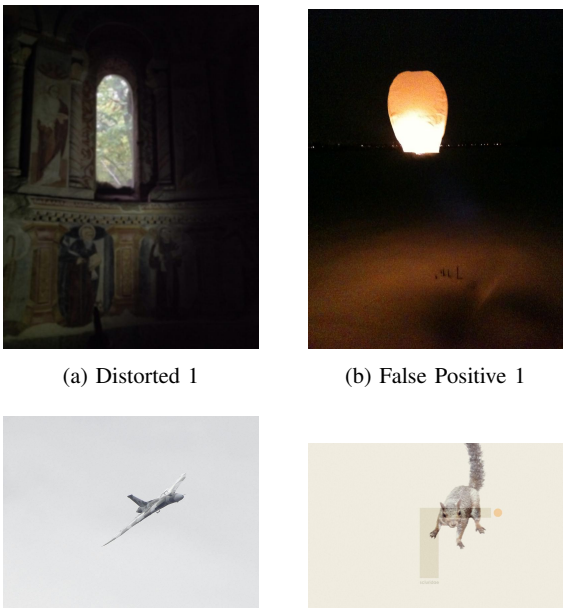
In summary, ViTHash is outperformed by existing perceptual hashing methods when exposed to visual quality distortions, yet it outperforms the existing methods on spatial distortions. This suggests that ViTHash has a higher resilience to changes in spatial orientation, which is a significant advantage in scenarios where images may be rotated, mirrored, or cropped.

Qualitative Results

To better understand where the strengths of ViTHash lie, this subsection delves into the mistakes made by ViTHash, aiming to gain deeper insight into what each method perceives as similar. The goal of this analysis is to assess how closely these methods align with human perception of similarity. In this part, the reporting is limited to only ViTHash and pHash, since the analysis is similar for pHash, dHash and PDQHash.

False positives are the most prevalent type of mistakes observed across the methods. In **Figure 5** and **Figure 6**, false positives are randomly selected to highlight these errors, for pHash and ViTHash, respectively. As seen in Figure 5, pHash produces false positives where the structural components of the images are somewhat similar, yet the actual content is completely different. For instance, matching a squirrel with an airplane is clearly mismatched from a human visual perspective. In contrast, Figure 6 shows that ViTHash, while yielding more false positives overall, tends to generate matches that can be considered visually similar from a human standpoint. This suggests that ViTHash aligns better with human perception in terms of what qualifies as visually similar content.

To better understand how pHash and ViTHash interpret similarity, an image that was not part of the original database was taken and the most similar matches were searched for. Both pHash and ViTHash had access to the exact same database for this comparison. **Figure 7** shows the target image, and the closest matches as identified by pHash and ViTHash. For pHash, the images deemed most similar would never be paired together by a human observer, as their content is vastly different. On the other hand, ViTHash's matches make more sense from a human perspective, as the closest matches show an insect similar to that in the target image.



(a) Distorted 1 (b) False Positive 1

(c) Distorted 2 (d) False Positive 2

Figure 5: Examples of a distorted images with their false positives for pHash. The false-positive images share the same high-level structure but show different content.

(a) Distorted 1

(b) False Positive 1

(c) Distorted 2

(d) False Positive 2

Figure 6: Examples of a distorted images with their false positives for ViTHash. The false-positive images share the same content, and therefore better aligns with human perception.

Visually Similar Images With A Different Acquisition Origin

The previous evaluations focused on mapping distorted images back to their original source. However, that evaluation alone does not fully capture ViTHash's ability to perceive visual similarity in a way that aligns with human perception.

To explore this further, scenarios are considered where images are visually similar but do not have the same acquisition origin. For example, taking two photographs of the same object from slightly different angles. A dataset consisting of 100 pairs of such images was created. An example of one such pair is shown in Figure 1.

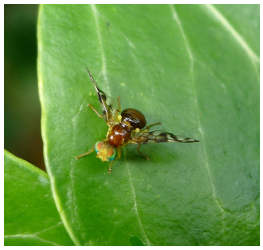
The pairs were split by placing one half of each pair into the query set and the other half into a database. The similarities between each query image and all images in the database were then computed. The distribution of similarity scores between matching pairs (i.e., the respective other half of the original pair) and non-matching pairs (i.e., all other images) was analyzed. As such, the aim is to assess each method's ability to detect visual similarity beyond identical source images.

Figure 8a and **Figure 8b** show the similarity score distributions for pHash and ViTHash, respectively. In the case of pHash (Figure 8a), the distributions for matching and non-matching pairs are almost identical. This indicates that pHash does not assign higher similarity scores to visually similar images from a different acquisition origin, making it ineffective in identifying such pairs.

(a) Target Image

(b) pHash Match 1

(c) pHash Match 2



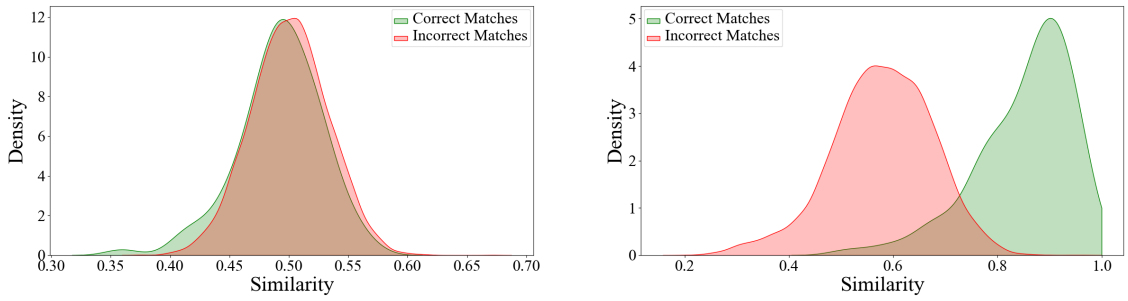
(d) ViTHash Match 1

(e) ViTHash Match 2

Figure 7: Similarity assessment for the target image. The first row shows the target image, followed by two rows depicting the images perceived as the most similar by pHash and ViTHash, respectively. Whereas the pHash closest matches show completely different content, the ViTHash closest matches both show an insect and therefore better align with human perception.

Conversely, ViTHash shows a clear distinction between the distributions (in Figure 8b). The similarity scores for matching pairs are noticeably higher than those for non-matching pairs. This demonstrates ViTHash's ability to capture human-like perceptions of visual similarity, effectively identifying images that are similar in content despite originating from different sources.

This experiment highlights that ViTHash's quantitative metrics may not fully reflect its capabilities. It excels in scenarios requiring a nuanced understanding of visual similarity. ViTHash aligns more closely with human perception, making it a valuable tool for ap-



(a) pHash (b) ViTHash

Figure 8: Similarity score distributions between query images and database images for (a) pHash and (b) ViTHash. For pHash, the distributions for matching pairs and non-matching pairs overlap significantly, demonstrating that pHash cannot identify visually similar images with a different acquisition origin. In contrast, for ViTHash, there is a clear separation between matching pairs and non-matching pairs, demonstrating that ViTHash is able to identify visually similar images with a different acquisition origin.

plications where identifying visually similar content is essential, even when images are not exact copies or originate from the same source.

FUTURE WORK & DISCUSSION

This section looks ahead by exploring several directions for future research, and discusses practical real-world applications.

Robustness

Visual quality distortions like blurring and compression have an impact on the robustness performance of ViTHash. Therefore, this can be studied further. In fact, this performance sensitivity should not come as a surprise, as blurring leads to a loss of (texture) features which will make it more difficult for the CLIP model to classify or recognise images accurately (due to possible ambiguity and feature misalignment in the latent space). The application of deblurring methods to restore or enhance the image quality before or during rescaling may be an interesting option for future work. Note that the rescaling step in Figure 2 was applied blindly to any image input into ViTHash. The fact that JPEG compression introduces sensitivities similar to blur may be due to the fact that it generally also acts as a low-pass filter, whilst introducing additional blocking artifacts etc. Alternatively, an interesting area for future research could be to use data augmentation techniques during training, or developing specialized loss functions that prioritize robustness to these distortions. As also discussed in the Integration in Real-World Applications section below, it may additionally be relevant to provide, e.g., blur or compression quality estimation values to weighted ranking and visualisation methods, workflow prioritisation strategies, or even presenting these estimations to the end-user.

Computational Cost

To achieve real-time or near-real-time applications, model compression techniques [21], software and hardware performance profiling and acceleration [22], [23], [24] and alternative ViT architectures could be explored [25]. Additionally, reducing the size of ViTHash hashes could help to optimize computational efficiency, as well as storage requirements. Moreover, utilizing the CLIP embedding space in a new architecture with a (lightweight) trainable head for image similarity detection may enhance performance and resilience to various image distortions.

Related Deep-Learning Work

Future work could also focus on exploring if other existing related work could improve or extend an overall hashing and deduplication framework. Such earlier work includes, e.g., the use of deep learning, CNNs and Siamese networks for implementing image comparison and deduplication strategies; see, e.g., [26], [27], [28], [29]. One notable advantage of using the ViTHash based approach over these methods will remain its direct relation to and its capability of also offering or facilitating text-based searches.

Integration in Real-World Applications

The application of traditional perceptual hashing methods has been used extensively by Internet Service Providers and government organisations for fighting CSAM. More recently, both technical [30] and legislative initiatives [31] have been extending this to, e.g., fighting extremism and terrorism. In fact, besides content moderation workflows, overall semantic searching, triage and deduplication tasks within the broader field of digital forensics are a major area of application for ViTHash and related technologies. In particular, the focus on numerical performance scoring of the methods may become less important, or would need to consider more advanced ranking performance scoring and (GUI) visualisation methods (not only considering first or top-ranking matches). Additionally, it is important to consider that, e.g., in an investigative context, the overall end-user problem is one of prioritisation in a human-controlled workflow. In turn, this prioritisation and results management typically requires suitable user interfaces for supporting human oversight and decision making. For example, such (LEA) tools, implementing automated ruleset-based and other forms of prioritisation, complemented by human-in-the-loop case management have been developed in [32] and [33]. Traditional binary, perceptual and now also AI- or ML-based hashing methods will most likely remain key technical components for driving the decision-making process within such prioritisation and result management frameworks. As these technologies continue to evolve, their integration into real-world applications will benefit from advancements in AI-driven hashing, ultimately enhancing digital forensics, content moderation, and investigative workflows.

CONCLUSION

This study explored the effectiveness of using Vision Transformers, specifically the ViT-L/14 architecture within the CLIP model, for image similarity detection in the context of identifying illicit content. Traditional perceptual hashing methods like pHash, dHash, and PDQHash have limitations in handling images that have undergone spatial transformations and are unable to identify images that are visually similar but have different acquisition origin. The proposed ViTHash method addresses these challenges by capturing a more nuanced representation of images that aligns closely with human perception.

Quantitative results showed that ViTHash outperforms traditional methods in detecting images subjected to spatial distortions, whereas it is outperformed by traditional methods for images with visual quality distortions. Qualitative analysis revealed ViTHash's strength in identifying visually similar images with a different acquisition origin. ViTHash's ability to recognize such similarities is particularly valuable for law enforcement agencies that are dealing with large volumes of data and that need to identify modified or related illicit content most effectively.

While ViTHash offers significant advancements in image similarity detection through its perception-aligned approach, it is not yet ready to replace traditional perceptual hashing methods. The limitations in computational efficiency and sensitivity to certain distortions suggest that a hybrid approach could be beneficial. Combining ViTHash with traditional methods may leverage the strengths of both, improving overall detection capabilities.

Future work should address ViTHash's sensitivity to visual distortions (such as blurring and JPEG compression) and its computational cost to enable real-time applications. Moreover, integrating insights from related deep-learning and deduplication frameworks could further enhance its real-world forensic capabilities.

By integrating further improvements and potentially combining ViTHash with existing methods, we expect that future research will lead to more robust tools, as such supporting agencies in combating the spread of harmful content, ultimately contributing to more effective interventions and reduced investigator workload.

Acknowledgment

This work was funded by the Research Foundation – Flanders (FWO), IDLab (Ghent University – imec), Flanders Innovation & Entrepreneurship (VLAIO), and the Flemish Government. NICC's participation to this paper was co-financed by the European Union (Belgian Internal Security Fund project, ISF-084-108).

Funded by
the European Union

REFERENCES

1. National Center for Missing & Exploited Children, "Cybertipline report 2023," Tech. Rep., 2023, last accessed on 23 October 2024. [Online]. Available: https://www.missingkids.org/content/dam/missingkids/pdfs/2023-CyberTipline-Report.pdf
2. Z. Cao and M. Zhu, "An efficient video similarity search algorithm," *IEEE Transactions on Consumer Electronics*, vol. 56, no. 2, pp. 751–755, 2010.
3. P. Corcoran and G. Costache, "Automated sorting of consumer image collections using face and peripheral region image classifiers," *IEEE Transactions on Consumer Electronics*, vol. 51, no. 3, pp. 747–754, 2005.
4. Q. Hao, L. Luo, S. T. Jan, and G. Wang, "It's not what it looks like: Manipulating perceptual hashing based applications," in *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*, 2021, pp. 69–85.
5. C. Hu, F. Yang, X. Xing, H. Liu, T. Xiang, and H. Xia, "Two robust perceptual image hashing schemes based on discrete wavelet transform," *IEEE Transactions on Consumer Electronics*, pp. 1–1, 2024.
6. T. Liu, H. Yao, X. Li, and C. Qin, "Robust perceptual image hashing for screen-shooting attack," *IEEE Transactions on Consumer Electronics*, vol. 70, no. 1, pp. 4367–4378, 2024.
7. S. Jain, A.-M. Cretu, and Y.-A. de Montjoye, "Adversarial detection avoidance attacks: Evaluating the robustness of perceptual hashing-based client-side scanning," in *31st USENIX Security Symposium 2022*. Boston, MA: USENIX Association, Aug. 2022, pp. 2317–2334. [Online]. Available: https://www.usenix.org/conference/usenixsecurity22/presentation/jain
8. J. De Geest, P. De Smet, L. Bonetto, P. Lambert, G. Van Wallendael, and H. Mareen, "Perceptual hashing using pretrained vision transformers," in *IEEE Gaming, Entertainment, and Media Conference (GEM) 2024*. IEEE, 2024, pp. 19–24. [Online]. Available: http://doi.org/10.1109/gem61861.2024.10585453
9. R. Sobti and G. Geetha, "Cryptographic hash functions: a review," *International Journal of Computer Science Issues (IJCSI)*, vol. 9, no. 2, p. 461, 2012.
10. J. Takeshita, R. Karl, and T. Jung, "Secure single-server nearly-identical image deduplication," in *2020 29th International Conference on Computer Communications and Networks (ICCCN)*, 2020, pp. 1–6.
11. R. Mehta, N. Kapoor, S. Sourav, and R. Shorey, "Decentralised image sharing and copyright protection using blockchain and perceptual hashes," in *2019 11th International Conference on Communication Systems & Networks (COMSNETS)*, 2019, pp. 1–6.
12. H. Farid, "An overview of perceptual hashing," *Journal of Online Trust and Safety*, vol. 1, no. 1, Oct. 2021. [Online]. Available: https://tsjournal.org/index.php/jots/article/view/24
13. C. Zauner, "Implementation and benchmarking of perceptual image hash functions," Master's thesis, Upper Austria University of Applied Sciences, 2010, available at http://phash.org/docs/pubs/thesis_zauner.pdf.
14. M. Fei, Z. Ju, X. Zhen, and J. Li, "Real-time visual tracking based on improved perceptual hashing," *Multimedia Tools and Applications*, vol. 76, pp. 4617–4634, 2 2017. [Online]. Available: https://link.springer.com/article/10.1007/s11042-016-3723-5
15. N. Krawetz, "Kind of like that - the hacker factor blog," Jan 2013, last accessed: 23 October 2024. [Online]. Available: https://www.hackerfactor.com/blog/index.php?/archives/529-Kind-of-Like-That.html
16. Meta, "Github PDQ," 2020, last accessed: 23 October 2024. [Online]. Available: https://github.com/facebook/ThreatExchange/tree/main/pdq
17. 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 *International conference on machine learning*. PMLR, 2021, pp. 8748–8763.
18. K. Han, Y. Wang, H. Chen, X. Chen, J. Guo, Z. Liu, Y. Tang, A. Xiao, C. Xu, Y. Xu, Z. Yang, Y. Zhang, and D. Tao, "A survey on vision transformer," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 1, pp. 87–110, 2023.
19. A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., vol. 30. Curran Associates, Inc., 2017. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf
20. Meta Research, "Github ISC 2021," 2021, last accessed: 23 October 2024. [Online]. Available: https://github.com/facebookresearch/isc2021
21. J. Zhang and et al., "Minivit: Compressing vision transformers with weight multiplexing," in *2022 IEEE/CVF Conference on Computer Vision and*

Pattern Recognition (CVPR), New Orleans, LA, USA, 2022, pp. 12 135–12 144. [Online]. Available: https://ieeexplore.ieee.org/document/9879562
22. H. Shi, X. Cheng, W. Mao, and Z. Wang, "P2-vit: Power-of-two post-training quantization and acceleration for fully quantized vision transformer," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 32, no. 9, pp. 1704–1717, Sept. 2024.
23. Z. Hou and S.-Y. Kung, "Multi-dimensional model compression of vision transformer," in *2022 IEEE International Conference on Multimedia and Expo (ICME)*, Taipei, Taiwan, 2022, pp. 01–06. [Online]. Available: https://ieeexplore.ieee.org/document/9859786
24. H. Song, Y. Wang, M. Wang, and Z. Wang, "Ucvit: Hardware-friendly vision transformer via unified compression," in *2022 IEEE International Symposium on Circuits and Systems (ISCAS)*, Austin, TX, USA, 2022, pp. 2022–2026. [Online]. Available: https://ieeexplore.ieee.org/document/9937660
25. Y. Fang, X. Wang, R. Wu, and W. Liu, "What makes for hierarchical vision transformer?" *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 10, pp. 12 714–12 720, Oct. 2023. [Online]. Available: https://ieeexplore.ieee.org/document/10147250
26. R. Kaur, J. Bhattacharya, and I. Chana, "Deep cnn based online image deduplication technique for cloud storage system," *Multimedia Tools and Applications*, vol. 81, pp. 40 793–40 826, 2022. [Online]. Available: https://link.springer.com/article/10.1007/s11042-022-13182-7
27. J. Melekhov, J. Kannala, and E. Rahtu, "Siamese network features for image matching," in *2016 23rd International Conference on Pattern Recognition (ICPR)*, Cancun, Mexico, 2016, pp. 378–383. [Online]. Available: https://ieeexplore.ieee.org/document/7899663
28. W. Hu, Y. Fan, J. Xing, L. Sun, Z. Cai, and S. Maybank, "Deep constrained siamese hash coding network and load-balanced locality-sensitive hashing for near duplicate image detection," *IEEE Transactions on Image Processing*, vol. 27, no. 9, pp. 4452–4464, 2018.
29. S. Kim, S. Min, W. Kim, D. Kim, and D. Hwang, "Enhancing deep hashing with gcn-based models for efficient similarity search," *IEEE Access*, vol. 12, pp. 187 278–187 289, 2024.
30. Europol, "How is europol keeping online spaces safe?" [Online]. Available: https://www.europol.europa.eu/media-press/newsroom/news/how-europol-keeping-online-spaces-safe
31. E. Parliament and C. of the European Union, "Regulation (eu) 2021/784 of the european parliament and of the council of 29 april 2021 on addressing the dissemination of terrorist content online." [Online]. Available: https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=celex%3A32021R0784
32. Aviator Project, "Why the automation of prioritization is needed," 2024. [Online]. Available: https://aviatorproject.com/articles/why-the-automation-of-prioritization-is-needed
33. GRACE Project, "Grace-fct project website," 2024. [Online]. Available: https://www.grace-fct.eu/

Jelle De Geest is a graduate with an M.S. degree in Computer Science Engineering with a major in Artificial Intelligence from Ghent University. This research was conducted as part of his master's dissertation at the IDLab research group. Contact him at jelledegeest@hotmail.be.

Patrick De Smet is a researcher and forensic expert at the NICC, Belgium. He currently holds the position of Chair of the ENFSI Digital Imaging Working Group. He is a member of the IEEE and the IEEE Computer Society. Contact him at patrick.desmet@just.fgov.be

Lucio Bonetto is assistant researcher at the National Institute of Criminalistics and Criminology. He's member of ENFSI Forensic IT Working Group. Contact him at lucio.bonetto@just.fgov.be

Peter Lambert is a full professor at IDLab, Ghent University — imec. His research interests include (mobile) multimedia applications, multimedia coding and adaptation technologies, and 3D graphics. He is a member of the IEEE and the IEEE Consumer Technology Society. Contact him at peter.lambert@ugent.be

Glenn Van Wallendael is an assistant professor at IDLab, Ghent University — imec. His main topics of interest are video compression including scalable video compression and transcoding. He is a member of the IEEE and the IEEE Consumer Technology Society. Contact him at glenn.vanwallendael@ugent.be.

Hannes Mareen is a postdoctoral researcher at IDLab, Ghent University — imec. His main areas

of interest are multimedia forensics, security and compression. He is a member of the IEEE and the IEEE Consumer Technology Society. Contact him at hannes.mareen@ugent.be.