Erschienen in: Proceedings of the 18th ACM/IEEE Joint Conference on Digital Libraries : JCDL '18. - New York, USA : ACM Press, 2018. - S. 131-140. - ISBN 978-1-4503-5178-2
https://dx.doi.org/10.1145/3197026.3197042

An Adaptive Image-based Plagiarism Detection Approach

Norman Meuschke, Christopher Gondek, Daniel Seebacher,
Corinna Breitinger, Daniel Keim, Bela Gipp
Department of Computer and Information Science
University of Konstanz, Germany
{first.last}@uni-konstanz.de

ABSTRACT

Identifying plagiarized content is a crucial task for educational and research institutions, funding agencies, and academic publishers. Plagiarism detection systems available for productive use reliably identify copied text, or near-copies of text, but often fail to detect disguised forms of academic plagiarism, such as paraphrases, translations, and idea plagiarism. To improve the detection capabilities for disguised forms of academic plagiarism, we analyze the images in academic documents as text-independent features. We propose an adaptive, scalable, and extensible image-based plagiarism detection approach suitable for analyzing a wide range of image similarities that we observed in academic documents. The proposed detection approach integrates established image analysis methods, such as perceptual hashing, with newly developed similarity assessments for images, such as ratio hashing and position-aware OCR text matching. We evaluate our approach using 15 image pairs that are representative of the spectrum of image similarity we observed in alleged and confirmed cases of academic plagiarism. We embed the test cases in a collection of 4,500 related images from academic texts. Our detection approach achieved a recall of 0.73 and a precision of 1. These results indicate that our image-based approach can complement other content-based feature analysis approaches to retrieve potential source documents for suspiciously similar content from large collections. We provide our code as open source to facilitate future research on image-based plagiarism detection.

KEYWORDS

Image Analysis, Plagiarism Detection, Academic Publishing

1 INTRODUCTION

Academic plagiarism has been defined as "the use of ideas, concepts, words, or structures without appropriately acknowledging the source to benefit in a setting where originality is expected" [4]. Forms of academic plagiarism vary in their degree of obfuscation ranging from unaltered copies (copy&paste), to slightly altered forms of plagiarism, such as interweaving text passages from multiple sources (shake&paste), to disguised forms of plagiarism, including paraphrases, translations, and idea plagiarism [27], and even the plagiarism of academic data [11].

The easily identifiable copy&paste-type plagiarism is more prevalent among students [14], while heavily modified plagiarism is more characteristic of researchers, who have strong incentives to avoid detection by skillfully disguising unoriginal content [2]. Research on plagiarism detection (PD) has yielded mature systems employing text retrieval to find similar documents. These systems reliably retrieve documents containing copied text, but often fail to identify disguised forms of academic plagiarism [27].

As we briefly explain in Section 2, several approaches have been introduced to complement text-matching methods and to improve the detection capabilities for disguised forms of plagiarism. Compared to the many sophisticated text-based retrieval approaches that have been proposed for PD, analyzing images to detect academic plagiarism has attracted little research. In this paper, we examine the use of image similarity detection techniques as a promising method for plagiarism detection when textual similarity is lacking.

For our use case, we define 'images' as the visual representations of data, e.g., in the form of bar charts, scatter plots, graphs, etc., as well as of concepts in the form of figures showing the schematic representations of entities and their relations, e.g., flow charts, organigrams, and component diagrams. Our definition also includes photographs and photo-realistic renderings.

Images enable conveying much information in a compressed format, and they represent this information differently from the information conveyed in text. These characteristics make images a promising feature to examine when assessing the semantic similarity present in academic documents. Identifying semantic similarity is crucial for detecting translated plagiarism and idea plagiarism. In some cases, even the plagiarism of data becomes detectable if the data values can be reconstructed from graphs.

The paper is structured as follows. In Section 2, we briefly present general PD approaches and previous work on image-based PD. We then begin Section 3 by informing our image-based PD approach through an investigation of image similarities found in documents that have been accused of constituting academic plagiarism. The remainder of Section 3 introduces the methods we developed and subsequently integrated into an adaptive and scalable image-based

PD approach capable of targeting the identified types of image similarity. In Section 4, we evaluate the image-based PD approach with respect to the types of image similarity we defined. We discuss our findings and present future work to be investigated in Section 5, before summarizing our work in Section 6.

2 RELATED WORK

2.1 Plagiarism Detection Approaches

Plagiarism detection is a specialized Information Retrieval (IR) task with the objective of comparing an input document to a large collection and retrieving all documents exhibiting similarities above a predefined threshold. PD systems typically follow a two-stage process consisting of candidate retrieval and detailed comparison [24]. For candidate retrieval, the systems commonly employ efficient text retrieval methods, such as n-gram fingerprinting or vector space models [15, 26]. For the detailed comparison, the systems typically apply exhaustive string matching. However, such approaches are limited to finding near copies of a text. To detect disguised forms of academic plagiarism, researchers have proposed a variety of mono-lingual text analysis approaches employing semantic and syntactic features, as well as cross-lingual IR methods [2].

Researchers also showed that hybrid approaches, i.e., the combined analysis of text and other content features, improve the retrieval effectiveness for PD tasks. Alzahrani et al. combined an analysis of text similarity and structural similarity [1]. Gipp and Meuschke showed that the combined analysis of citation patterns and text similarity improves the identification of concealed academic plagiarism [5, 16]. Pertile et al. confirmed the positive effect of combining citation and text analysis and devised a hybrid approach using machine learning [20]. Recently, Meuschke et al. demonstrated the benefit of analyzing the similarity of mathematical expressions [17] and patterns of semantic concepts [18] for improving the identification of academic plagiarism.

2.2 Image Analysis for Plagiarism Detection

Few studies have investigated the analysis of image similarity for PD. Hurtik and Hodakova use higher degree F-transform to provide a highly efficient and reliable method to identify exact copies of photographs or cropped parts thereof [8]. However, the method does not consider image alterations aside from cropping.

Iwanowski et al. evaluate the suitability of well-established feature point methods, such as SIFT, SURF, and BRISK, to retrieve exact and visually altered copies of photographs [9]. Srivastava et al. address the same task using a combination of SIFT features extracted using SIFT and perceptual hashing [23].

Feature point methods identify and match visually interesting areas of a scene. The methods are insensitive to affine image transformations, such as scaling or rotation, and relatively robust to changes in illumination or the introduction of noise.

Perceptual hashing describes a set of methods that map perceived content of images, videos, or audio files to a hash value (pHash) [7]. Images perceived as similar by humans also result in similar pHash values, in contrast to cryptographic hashing, in which a minor change in the input results in a drastically different hash value. Thus, the similarity of images can be quantified as the similarity of their pHash values. If image components, such as shapes, are re-arranged, both feature point methods and perceptual hashing often fail.

Iwanowski et al. mention that the effectiveness of the feature point approaches they tested decreases if the test images consist of multiple sub-images. We also observed this limitation in our tests. For example, the two compound images shown in Figure 10 in Appendix A consist of six and four sub-images, respectively. The image in the later document omits two of the sub-images present in the compound image from the source document. Applying the combination of SIFT feature extractor and MSAC feature estimator to compare these two compound images correctly identifies a high similarity between the two sub-images at the top in both compound images, but does not establish a similarity for the other sub-image pairs. This problem can be solved by decomposing the compound image into sub-images and applying near duplicate detection methods, such as perceptual hashing, as we show in our evaluation (cf. Case 6 in Table 1, Section 4).

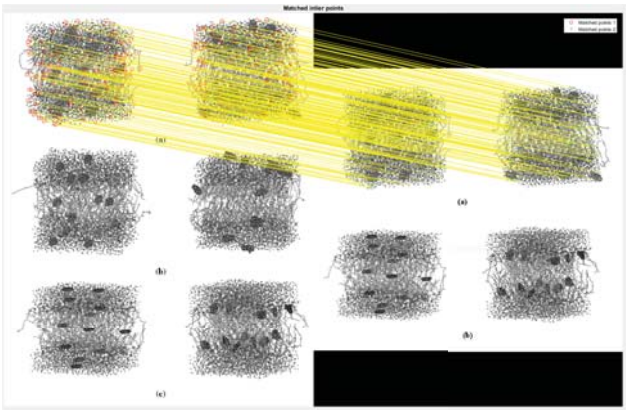

Figure 1: Comparison of compound images using SIFT+MSAC. The approach can only establish a similarity for some of the sub-images.

Feature point methods and perceptual hashing typically also fail to establish meaningful similarities for images primarily containing text, e.g., tables inserted as images. Typically, the feature points for individual letters are matched to multiple letters occurring in different places in the comparison document, which prevents identifying meaningful clusters of matching features.

In summary, prior research on image-based PD proposed methods that reliably retrieve exact and cropped image copies and images that underwent affine transformations. These methods focus on photographs, for which they achieve good results even if photo quality is reduced or modified, e.g., by blurring.

For images that underwent other modifications, such as rearranging shapes in the image, redrawing components of the image, or for images that consist primarily of text, the proposed methods often fail. Compound images should be split into meaningful sub-images before applying feature point methods or perceptual hashing to achieve the best retrieval performance. Identifying other types of image similarity than the comparably modest alterations detectable with the approaches we presented in this section requires additional use-case-specific analysis approaches.

3 ADAPTIVE IMAGE-BASED PD

The studies on image-based PD presented in Section 2.2 focused on providing reliable and computationally efficient methods to identify exact copies and cropped image copies, as well as affine image transformations. Our goal is to offer an efficient detection approach capable of identifying a larger subset of potentially suspicious image similarities. Specifically, we seek to enable the detection of similar images found in academic documents. To derive the requirements for such a detection approach, we examined alleged and confirmed cases of plagiarism, as we explain in the following section.

3.1 Types of Image Similarity

We used the VroniPlag collection[1] as a source for real-world cases of similar images in academic documents. The VroniPlag project (based in Germany) is a crowd-sourced effort investigating plagiarism allegations. Most of the examined works are doctoral dissertations written in German or English. Each document has been manually examined for potential plagiarism and has been annotated by several users according to a standardized workflow.

The project documents all findings, discussions, and other information pertaining to each examined case in the form of a wiki. Each passage of the analyzed document, for which similarities to a source document have been identified, is documented as a 'fragment' using a dedicated page in the wiki. Each fragment has been independently examined by a minimum of two users. Each work contains anywhere from a few dozen to several hundred fragments.

The analyses by the VroniPlag project yield a high-quality annotated dataset, which is continuously being expanded with new cases. At the time of writing, the dataset was composed of 196 academic works containing alleged instances of plagiarism[2]. For most cases, the responsible universities confirmed breaches of academic integrity resulting in the withdrawal of doctorates or other sanctions, such grade reduction or a formal reprimand. In several cases, official investigations are pending. In 14 cases, no official investigation has been initiated, e.g., because the statute of limitation for the alleged offense had passed[3].

Using a targeted Web crawler, we retrieved all pages of the VroniPlag wiki documenting fragments that involve the use of similar images. Hereafter, we describe and classify into broader categories the types of image similarity we observed during our review of these fragments.

3.1.1 Exact Image Copies. We define two images as exact copies if they have identical dimensions and the values and positions of their pixels match. This type of image similarity is very rare, since authors who reuse images are usually not able to access the original image file. In our investigation of the VroniPlag collection, no cases of exact copies were found. Image cropping or changes that are inadvertently introduced when authors reuse images from a PDF or print version of the source document are the main reason why exact image copies are extremely rare. Copying digital images from a PDF document will typically re-compress the images, resulting in rearranged pixels and the loss of information.

3.1.2 Near Image Copies. We classify images as near copies if they share the large majority of their visual content, yet exhibit minor differences introduced by *i)* removing non-essential content (e.g., numeric labels or watermarks), *ii)* cropping or padding, *iii)* performing affine transformations (e.g., scaling or rotation), *iv)* changing the resolution, contrast or color space. Especially changes of the categories *iii)* and *iv)* can be introduced inadvertently by extracting and reusing images from a PDF or printed document.

We frequently encountered near image copies in our investigation. Figure 5 in Appendix A is a representative example. The author reused an illustration of a kidney from Wikipedia[4] without attribution. Some lines that connect labels to points in the illustration have been removed in the reused image.

3.1.3 Altered Image Reuse. We define altered image reuse as containing differences that required purposeful actions to visually change the reused image. Altered image reuse is hard to classify conclusively given the virtually infinite possibilities for modifying a visual representation. Given our observations, we distinguish three broad levels of alteration.

i) Weakly altered images typically reuse parts of an original image as near copies. Figure 10 in Appendix A shows an example in which sub-images of a compound image were reused.

ii) Moderately altered images typically reuse most or all the visual components of the original image, yet significantly rearrange the components. Figure 12 in Appendix A shows a typical example for moderately altered image reuse.

iii) Strongly altered images are typically completely redrawn versions of the source with significant changes made to the arrangement and/or visual appearance of image components. Figure 14 in Appendix A shows an example of this type of alteration. The two technical drawings show construction plans with identical dimensions, yet the arrangement of the sub-images and the placement of the labels and measurements differ.

3.1.4 Visualizing Reused Data. Reusing data or the visualizations of data without correct attribution may constitute plagiarism or data fabrication if the data presented was not collected. Figure 18 and Figure 19 in Appendix A show near-identical bar charts and line charts in the VroniPlag collection. We found no cases in which reused data was visualized differently. However, given that misuse of data is a well-known problem in academia [11], we hypothesize that such cases do exist, and we believe that our image-based PD analysis will contribute to making them identifiable.

3.2 Requirements Analysis

Given the types of image similarity we observed in the VroniPlag collection, we derive the following requirements for methods to detect such similarities.

Most similar images we observed fell into the category of near copies. This result was to be expected, since investigations of plagiarism allegations exhibit a known bias towards identifying less obfuscated, hence easier to spot, forms of content reuse [21]. The effort necessary to detect disguised forms of content reuse and the lack of tools to support users with that task result in a lower probability of discovering disguised plagiarism instances.

[1] http://de.vroniplag.wikia.com/wiki/Home
[2] http://de.vroniplag.wikia.com/wiki/VroniPlag_Wiki:Statistik
[3] http://de.vroniplag.wikia.com/wiki/VroniPlag_Wiki:Aberkennungen
[4] https://commons.wikimedia.org/wiki/File:Kidney_PioM.png

Retrieving near copy images requires methods to reliably identify and efficiently match visually apparent features. Obtaining a semantic understanding of the image composition and underlying data, e.g., by incorporating knowledge about the image type, is typically unnecessary. Robustness against minor, potentially unintentional variations in image quality and dimensionality are important. Another requirement is computational efficiency, since the PD task requires comparing documents to large collections.

The task of retrieving slightly altered images can often be reduced to identifying near copies of image sections. For such cases, methods for identifying sub-images are key to achieving a high retrieval effectiveness. Detecting moderately and strongly altered images often requires obtaining a deeper semantic understanding of the data being visualized. Since the visual appearance of the data differs, additional features, such as labels, as well as information from the text surrounding the images, should be considered.

Employing image analysis to identify data reuse is a challenging retrieval task, since it requires bridging the semantic gap between the visual representation and the underlying data. We expect that identifying visually different representations of (near) identical data requires methods tailored to analyzing specific types of visualizations, such as bar charts, box plots, line graphs, or scatter plots. Similarly to the way in which humans interpret data visualizations, the methods should consider all available information, e.g., size, shape, color, and position of data points, axes scales, as well as the content and position of labels and legend entries.

Due to the variety of possible image similarities, we regard a combination of multiple analysis methods as most promising for covering the spectrum of similarities. The next section describes the detection approach we developed to address these requirements and the insights from prior work (cf. Section 2).

3.3 Overview of the Detection Process

Figure 2 illustrates the adaptive image-based detection process, for which we describe the key components in the following subsections.

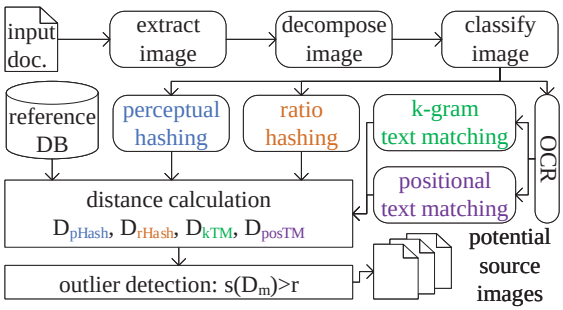

Figure 2: Overview of adaptive image-based PD approach.

The input to the process is a PDF, from which we extract the contained images and check whether they contain meaningful sub-images (cf. Section 3.4). To reduce the computational load for the system, we use the convolutional neural network (CNN) described in Section 3.5. The CNN classifies images according to their suitability for being analyzed using the different analysis methods.

Currently, our approach includes four analysis methods to identify potentially suspicious image similarity. We employ perceptual hashing as a well-established, fast, and reliable method to find highly similar images (cf. Section 3.6). To improve the identification of disguised image similarity, we employ two approaches that perform text-matching for the text extracted from images using Optical Character Recognition (cf. Section 3.7 for our OCR approach). The first approach, k-gram text matching (cf. Section 3.8), is a well-established, text-based PD method. The second approach, positional text matching (cf. Section 3.9), is a contribution of this paper. Another contribution is the ratio hashing approach (cf. Section 3.10), which we propose to identify highly similar bar charts. Ratio hashing is a first, specialized method to identify data reuse.

All analysis methods are applied independently of each other. The methods compute method-specific feature descriptors, which they compare to the method-specific feature descriptors for all documents of the reference collection. We use a relational database to store feature descriptors. The comparisons of feature descriptors yield separate lists of distance scores D_m for each analysis method m. The lists are ordered in ascending order of the distance scores and provided as the input to an outlier detection process (cf. Section 3.11). The outlier detection process computes method-specific suspiciousness scores $s(D_m)$ that indicate whether clear outliers exist within the lists of method-specific distance scores. The process then returns as potential sources for an image in the input document all images for which at least one method-specific suspiciousness score $s(D_m)$ is larger than a reporting threshold r.

We implemented the detection process as an Python 2.7 application that handles all inputs and outputs via a command line interface. This setup allows for easy integration of the application into existing IR systems as a loosely coupled module. The code is open source and available at:

www.purl.org/imagepd

3.4 Image Extraction and Decomposition

To extract the images contained in the input documents, we use poppler[5], an open source library for PDF processing. To reduce storage requirements, all images are converted to JPEG. To reduce computational effort and avoid false positives, we discard JPEG images with a file size below 7,500 bytes. This threshold reflects our observation that images with less than 7,500 bytes typically contain single characters, logos, or decorative elements that are of little value for identifying potential instances of plagiarism.

To decompose compound images, such as Figure 10 in Appendix A, we devised a heuristic process based on two assumptions. First, we assume that white pixels separate sub-images. Second, we assume that sub-images are rectangular and aligned horizontally or vertically within the compound image. Although these assumptions exclude some images, we consider the approach a reasonable trade-off between accuracy and computational effort. If successful, image decomposition can increase the detection performance for sub-images (cf. Section 2.2). However, compound images, for which image decomposition fails, are still analyzable.

The decomposition process includes the following steps: *i) conversion to grayscale* to reduce runtime; *ii) padding with white pixels* to remove a potential border; *iii) binarization using adaptive thresholding* to obtain a black and white image; *iv) dilation* to ensure

[5]https://poppler.freedesktop.org/

black pixels are connected; *v) floodfill* of white areas with black pixels; *vi) subtract original image*; *vii) invert image*; *viii) blob detection* using the algorithm of Suzuki and Abe [25]; *ix) estimate bounding box* by looking for large contours aligned along the image axes; *x) crop and store the identified sub-images in the reference database.*

3.5 Image Classification

We use a deep convolutional neural network to distinguish photographs and bar charts from other image types. Ratio hashing is exclusively applied to bar charts. Photographs are exclusively analyzed using perceptual hashing, since they typically contain too little text to apply OCR text-matching. All other image types are analyzed using perceptual hashing and OCR text matching.

The CNN implements the AlexNet architecture [13]. We used the Caffe framework [10] to train the CNN. Manually checking 100 classified images showed that the CNN achieves an accuracy of 92% for photographs and 100% for bar charts.

3.6 Perceptual Hashing

We include perceptual hashing in the detection process, since prior research demonstrated the suitability of the approach to reliably retrieve near copy images [7, 23]. In the experiments of Srivastava et al., perceptual hashing achieved an accuracy of 0.84, which was the second best result following SIFT, which achieved an accuracy of 0.95. Given that SIFT required approx. four times longer runtime than perceptual hashing, and perceptual hashing outperformed other prominent feature point methods, such as SURF, FREAK and KAZE [23], we consider the approach to be a reasonable trade-off between accuracy and computational complexity.

We tested different variants of perceptual hashing. We found that using a Discrete Cosine Transform (DCT) and comparing pHash values using their hamming distance achieved the best accuracy. The hamming distance of two pHash values is the number of bits that differ in the hashes. We precompute the pHash values for all images of the reference collection and store the pHash values in the reference database. In its current state, our prototype employs pairwise comparisons of the pHash for an input image to all pHash values in the reference database. To enable comparing an image to very large collections, the pairwise comparisons can be replaced with a locality sensitive hashing approach to speed up the process, as demonstrated by Srivastava et al. [23].

3.7 OCR Preprocessing

Including textual features in the similarity analysis requires a preprocessing step to extract the text from images. Research on OCR has provided a wide range of approaches for this task. We chose the open-source OCR engine Tesseract [22], because it allows extracting both characters and words, including their positions. Tesseract is widely-used, actively maintained, and repeatedly outperformed proprietary OCR engines in recognizing English texts [19].

Prior to applying Tesseract, the image is normalized to a height of 800 pixels while maintaining the aspect ratio, which significantly improves recognition. OCR is computationally expensive and processing times vary greatly depending on the input image. We extract the text for each image in the reference collection once and store the information in the reference database.

3.8 k-gram Matching

Determining textual similarity by analyzing matching word or character k-grams is a well-established IR approach. Numerous PD approaches employ variable-size or fixed-size k-grams [2, 15, 26]. For regular texts, k-grams with lengths corresponding to 3-5 words, i.e., approx. 15-30 characters, are used most frequently [3, 6, 12].

To choose a k-gram size for analyzing text in figures extracted using OCR, two use-case specific factors should be considered. First, images typically contain smaller text fragments, such as labels or bullet points. Second, we extract the text content of images using OCR, which is likely to introduce noise, i.e., wrongly recognized characters. Such recognition errors can significantly reduce the accuracy, especially for word k-gram approaches.

To account for the likelihood that incorrectly recognized characters occur, we chose a comparably fine-grained k-gram resolution of three characters. Given the typically sparse presence of text in figures, we retain all k-grams identified for an image as an unordered set that forms the k-gram descriptor of that image. Typically, k-gram-based PD approaches that analyze entire documents employ some form of k-gram selection [2, 15, 26]. We form the k-gram descriptor for all images of the reference collection during preprocessing and store the descriptors in the reference database.

Currently, our prototype performs pairwise comparisons of the k-gram descriptor of an input image to all k-gram descriptors of the reference collection. To scale the image-based detection approach to very large collections, an additional filtering step can easily be introduced, e.g., by indexing individual k-grams and requiring a minimum k-gram overlap to perform the full comparison of the k-gram descriptors. To quantify the distance d of two k-gram descriptors K_1 and K_2, we use the set-based distance function $d = \frac{K_1 \ominus K_2}{K_1 \cap K_2}$, in which $\ominus$ represents the symmetric difference.

3.9 Position-aware Text Matching

As explained previously, OCR errors are a serious threat to the retrieval performance of text-matching approaches. This problem is further aggravated by the typically sparse amount of text present in academic images. To improve the robustness of similarity assessments examining textual content in images, we propose including positional information as an additional feature of the analysis. Specifically, we suggest to only consider text matches for computing the similarity of two images if the matching text occurs in broadly similar regions in both images.

Figure 3 illustrates the approach. We assume an input image (left) and a comparison image (right) that have been scaled to the same height or width while maintaining the individual aspect ratio of the images. The markers A, B, C and D in Figure 3 symbolize text identified by the OCR engine, e.g., characters, words, or k-grams. Each text fragment identified in the input image is considered the center point around which a proximity region is defined. In Figure 3, a fixed-size circle is used as the proximity region, however other shapes and dynamic sizing of the shape, e.g., dependent on the length of the text fragment, are also possible. The proximity regions of the input image are projected into the comparison image. To compute the similarity score, only the text matches that also occur in corresponding proximity regions (A and D) are considered.

Text matches that occur outside of a defined proximity region (B) and non-matching text (C and X) are disregarded.

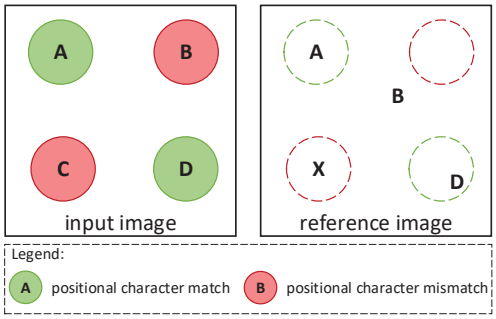

Figure 3: Illustration of position-aware text matching.

Our current prototype implements position-aware text matching using the following approach. Images are rescaled to a common height of 800px while maintaining their aspect ratio. We use single characters as the center points around which a fixed-size circular proximity region of 50px is defined. The distance function considers the number of position-aware text matches divided by the number of characters in the longer of the two OCR texts. This normalization reflects the assumption that two images are less likely to be similar if their amount of textual content differs strongly.

The pairwise comparison of position-aware text matches is computationally more expensive than the set-based k-gram comparison. For this initial study, we employ no filtering of candidate images except for the classification described in Section 3.5. To scale the approach, filtering heuristics, such as setting a minimum threshold for matching k-grams (cf. Section 3.8), can be added.

3.10 Ratio Hashing

To demonstrate a possible approach to target the plagiarism of academic data and results, we propose ratio hashing to identify semantically similar, yet visually differing bar charts. Due to the diversity of chart types, each type must be treated with a different approach. Since bar charts are very common in academic publications, ratio hashing is geared towards them. The idea of ratio hashing is to compute a hash value from the relative heights of bars compared to the height of the largest bar.

To extract the bar heights from an input image, we process the image as follows: *i) convert to grayscale*; *ii) binarize* using global thresholding to obtain a black and white image with sharp contours; *iii) pad image with white pixels* to ensure bars can be filled; *iv) clean artifacts* of black pixels using a threshold on the relative area covered by the pixels; *v) remove image border*; *vi) floodfill with black pixels and invert*; *vii) find candidates for bars* by determining the lengths of all vertical lines of black pixels, *vii) determine bars* by clustering vertical lines, remove noise from whiskers, labels, and legend entries, then assume the average height of the lines in a cluster as the bar height. Once the bar heights have been determined, we sort the bars by decreasing order of their height to speed up the comparison of two ratio hashes. We then calculate the relative bar heights and store the ratio hash, i.e., the sequence of relative bar heights, in the reference database.

To determine the distance between two ratio hashes, we compare the components of the hash, i.e., the relative bar heights, in decreasing order and calculate the sum of the differences of the bar heights. We currently compare the ratio hash of an input bar chart to all ratio hashes in the reference database. In the future, the computational effort of the approach can be reduced by indexing ratio hashes and implementing filtering steps. For our initial evaluation, we limited computational effort by requiring bar charts to have the same number of bars. However, the comparison approach can easily be changed to more exhaustive comparisons that consider the best fit between sets of different sizes if the analysis scope is reduced through prior filtering steps.

3.11 Outlier Detection

Each of the four analysis methods described in the previous subsections returns the method-specific distances of the input image to all images in the reference collection as an ordered list D_m. To quantify how suspicious, i.e., how indicative of potential image reuse, these distances are, we make two assumptions.

First, we assume that the input image can only be suspicious of being derived from (an)other image(s) in the collection if it exhibits *comparably strong similarities*, i.e., small distances, to a small number c of other images. Small distances of the input image to other images alone are not necessarily suspicious. The input image could be a logo that the preprocessing step missed to exclude. Such images would exhibit small distances to many other images in the collection. Therefore, we additionally require that the input image exhibits small distances to fewer than c other images. The cutoff parameter c is set to 10 in our system. In essence, c is a filter for false positives that accounts for potential deficits of the detection process or the collection, e.g., common images or multiple versions of documents not eliminated during preprocessing. The parameter should be chosen large enough to rule out any reasonable possibility that strong similarities to more than c documents are not false positives. We consider $c = 10$ a conservative estimate to ensure this property even for large collections.

Second, we assume that image similarities are comparably strong if *a clear separation* is observable in the distance scores for the $k < c$ images most similar to the input image and the distance scores of the remaining images in the collection. In other terms, images with strong similarities to the input images must be outliers. If that requirement is not fulfilled, the input image is either genuine or too dissimilar to a potential source for being detected. Alternatively, the reference collection may not contain the source image or the analysis method failed to determine a meaningful distance.

Given these assumptions, we detect outlier distances as depicted in Figure 4 and described hereafter. For each analysis method m, the absolute distances $d_i, 1 \leq i \leq n$ of the input image to all n images of the reference collection are stored as a list $D_m = (d_1, d_2, \ldots d_n)$ in ascending order of d_i. Each method-specific list of absolute distances D_m is transformed to a list D'_m of the relative deltas d'_i between d_i and d_{i+1} as follows: $d'_i = \frac{(d_{i+1} - d_i)}{d_i}, 0 < i < |D_m|, i \in N$.

To find outliers, i.e., elements in D'_m that exhibit a clear separation to succeeding elements, we sequentially scan through D'_m and

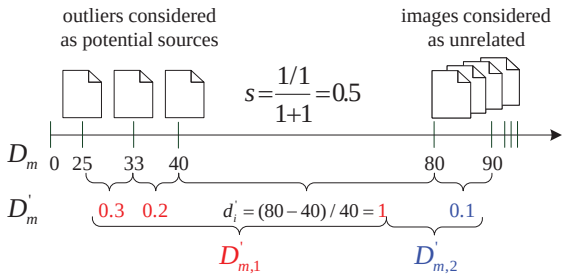

Figure 4: Illustration of the outlier detection process.

check for elements d'_i that exceed a threshold t. The threshold is customizable and set to $t = 1$ for all methods except for k-gram matching, where we found a threshold of $t = 2$ to yield better results. In other terms, we require that a pair of distances (d_i, d_{i+1}) exists, for which d_{i+1} is at least twice as large as d_i and three times as large as d_i in the case of k-gram matching. If an d'_i exceeds these thresholds, D'_m is split into $D'_{m,1}$ and $D'_{m,2}$ at the largest d'_i, where $d'_i \in D'_{m,1}$. If $D'_{m,1}$ has less than c elements, the corresponding images are considered potential sources of the input image.

The final suspiciousness scores $s(D_m)$ for each method-specific list of distances D_m are calculated as $s = \frac{\bar{d}}{1+\bar{d}}$ where $\bar{d} = \frac{\max(d'_i \in D'_{m,1})}{t}$. In other terms, s considers the relative delta in the distances that separates a previously determined group of outliers (in our case at most $c - 1 = 9$ images) from the remainder of the collection. We use the function $y = \frac{x}{x+1}$ to normalize the score s to $[0, 1]$. The sublinear normalization function assigns a weight of 0.5 if the image in the outlier group that is least similar to the input image is separated from the remainder of the collection by a margin that is as large as the absolute distance of this least similar outlier to the input image. For all analysis methods, we set $s = 0.5$ as the threshold to consider an image potentially suspicious and a score $s > 0.75$ as highly suspicious. For the case $s = 0.75$, the least similar outlier has a distance margin to the next similar image that is three times as large as its absolute distance to the input image.

4 EVALUATION

To evaluate the adaptive image-based PD process, we selected 15 image pairs from documents in the VroniPlag collection. We chose images that reflect the spectrum of image similarities we observed in the collection (cf. Section 3.1). We list the test cases, most of which are from the life science domain, in Appendix A. To create a realistic test collection, we obtained 4,500 random images contained in life science publications from the open access repository PubMed Central[6]. We hid the 15 known source images among the 4,500 obtained images and created the reference database by classifying each image and computing the applicable feature descriptors. After precomputing the reference database, we used each of the 15 reused images individually as input for the detection process.

Table 1 shows the method-specific suspiciousness scores $s(D_m)$ for each input image computed from the distance scores D_m of the four analysis methods, perceptual hashing (pHash), character trigram matching (OCR k-grams), position-aware character matching (OCR Pos.) and ratio hashing (rHash).

#	Image Type	Alteration	pHash	OCR k-grams	OCR Pos.	rHash
1	Illustration	near copy	**0.86**	< 0.5	< 0.5	-
2	Illustration	near copy	**1.00**	0.79	0.77	-
3	Illustration	near copy	**0.87**	< 0.5	< 0.5	-
4	Micr. Image	near copy	< 0.5	< 0.5	< 0.5	-
5	Table	near copy	< 0.5	< 0.5	< 0.5	-
6	Illustration	low	**0.78**	< 0.5	< 0.5	-
7	Illustration	low	**0.57**	< 0.5	< 0.5	-
8	Illustration	medium	< 0.5	**0.87**	< 0.5	-
9	Table	medium	0.62	**0.71**	0.55	-
10	Illustration	high	< 0.5	< 0.5	< 0.5	-
11	Table	high	< 0.5	**0.79**	< 0.5	-
12	Table	high	< 0.5	**0.92**	< 0.5	-
13	Line Chart	high	< 0.5	**0.70**	< 0.5	-
14	Bar Chart	near copy	0.62	0.64	0.77	**0.92**
15	Line Chart	near copy	< 0.5	< 0.5	< 0.5	-

Table 1: Suspiciousness scores for input images.

#	Image Type	Alteration	pHash	OCR k-grams	OCR Pos.	rHash
1	Illustration	near copy	1	> 10	> 10	-
2	Illustration	near copy	1	1	1	-
3	Illustration	near copy	1	> 10	> 10	-
4	Micr. Image	near copy	1	> 10	> 10	-
5	Table	near copy	> 10	> 10	> 10	-
6	Illustration	low	1	> 10	> 10	-
7	Illustration	low	1	> 10	> 10	-
8	Illustration	medium	1	1	> 10	-
9	Table	medium	1	1	1	-
10	Illustration	high	1	> 10	> 10	-
11	Table	high	> 10	1	> 10	-
12	Table	high	1	1	> 10	-
13	Line Chart	high	> 10	1	> 10	-
14	Bar Chart	near copy	1	1	1	1
15	Line Chart	near copy	> 10	> 10	> 10	-

Table 2: Ranks at which source images were retrieved.

Table 2 complements Table 1 by showing the ranks at which each of the four analysis methods retrieved the source image for an input image. Note that the system would not return any results for input images with a score below 0.5, as no similarities that form clear outliers were identified in such cases. To verify the appropriateness of this threshold, we retrieved for each of the input images having scores below 0.5 the 10 images identified as most similar and checked whether this set contained the source image. Limiting the set to 10 images is a heuristic that assumes a reviewer might be willing to browse through ten results, although none of them has been identified as clearly suspicious.

As shown in Table 1, scores above the reporting threshold of 0.5 were determined by at least one analysis method for 11 of the 15 input images, thus achieving a recall of 0.73. The cases 4, 5, 10, and 15 are false negatives, although pHash retrieved the source images for cases 4 (a microscope image) and 10 (a visually sparse sketch

[6]https://www.ncbi.nlm.nih.gov/pmc/tools/openftlist/

exclusively using basic geometric shapes) at the top rank. For these two cases, pHash computed similarities to many unrelated images of the same types. However, the low score assigned to the pHash distances shows that the identified similarities are not clear outliers. For case 5, the low quality of the input image caused insufficient OCR results to detect the nearly copied text in the table. The line chart in case 15 is visually too sparse to be detected by pHash. Little textual content and low image quality also caused the OCR-based methods to fail for this case.

As shown in Table 2, the true source images were retrieved at the top rank for all input images with a score above 0.5. For cases in which the method-specific score was lower than 0.5, no source image was retrieved among the top-ten most similar images. This result shows that the outlier detection process effectively eliminated all false positives, resulting in a precision of 1. A high precision is important for PD approaches to avoid false suspicion.

For near copies and weakly altered images, perceptual hashing in combination with sub-image extraction worked well, yielding suspiciously high scores for six of the nine cases falling into these categories. Text analysis utilizing OCR performed better than perceptual hashing for moderately and strongly altered images if the quality of the image was high enough to perform OCR reliably and if sufficient text content is present. The OCR-based approaches identified three of the four cases that involved tables (cases 9, 11, 12), for which they yielded clearly suspicious scores (0.71, 0.79, and 0.92, respectively). While k-gram matching performed better than the position-aware text matching for most cases, the position-aware text matching was more robust to low OCR quality. Combining both approaches therefore allows to process a larger number of input images in a real-world setting.

The test dataset contains only one case (14), in which a bar chart was reused. For this case, ratio hashing clearly outperformed all other methods (s=0.92), although the bar chart was rotated and slightly altered. Clearly, additional evaluations must be conducted to reliably determine the performance of ratio hashing.

Creating the reference database for the 4,515 images took around two hours using a desktop computer with a 2.70 GHz Intel Core i5-6400 CPU, 8 GB of main memory and a GeForce GTX960 GPU, which was used to accelerate the CNN classifier. Executing the analysis methods took between 1-3 seconds for perceptual hashing and ratio hashing and between 2 to 16 seconds for the OCR-based methods using the same computer. This time includes classifying the input images, computing the feature descriptors, and comparing the descriptors to all other descriptors in the database.

5 DISCUSSION & FUTURE WORK

Our results demonstrate that an adaptive image-based PD approach enables the identification of a wide range of suspicious image similarities in academic work. While the suitability of analysis methods strongly depends on the individual images, the combination of analysis methods achieved a good recall of 0.73 in our experiments. The proposed outlier detection process performed particularly well in our experiments. Using restrictive thresholds, our approach eliminated false positives and achieved a precision of 1.

While these results are promising, our small test collection, in which images from life science publications were overrepresented, limits the generalizability of our results. Future experiments must show whether the properties we require to assume that image similarities form suspicious outliers will also be observable in significantly larger collections. The outlier detection process mainly depends on the distribution of distances between an input image and the images in the reference collection. In large collections, unrelated images may exhibit similarities to the input images that prevent identifying clear outlier similarities. In such cases, reducing the suspiciousness threshold may become necessary and would likely result in the identification of false positives. Since the outlier detection process operates on a simple list of precomputed distance scores, adjusting the threshold at runtime is feasible. Hence, a frontend application could allow users to interactively adjust the threshold to determine the number of results (including potential false positives) the user is willing to examine.

Our approach is well suited to be scaled for an evaluation on much larger collections. All preprocessing steps can be executed in parallel. In the current implementation of the approach, only the time required for the comparison of feature descriptors depends on collection size. We described several easily implementable options to decrease the linear runtime requirement of this step by adding feature indexing and feature selection approaches.

However, an inherent challenge to conclusive, large-scale evaluations of PD approaches is the difficulty of compiling test collections. A widely accepted solution is to use collections containing artificially created plagiarism instances. For image plagiarism, such collections do not yet exist. Even if they existed, it would be questionable whether they are representative of the real-world plagiarism that is committed by experienced researchers with a strong incentive to hide their misconduct. For this reason, we opted to use real-world cases of image reuse in our experiments.

A technical challenge to the detection effectiveness of the proposed PD approach in the real-world detection settings we imposed for our experiments is OCR effectiveness. The OCR-based analysis methods showed the best results for medium to high-level alterations of images. However, poor image quality, especially for older digitized academic papers, reduces OCR performance. Perceptual hashing often performed poorly for visually sparse images. A dilation step might help achieve better results. Although our approach to sub-image extraction performed well for most cases, sometimes it failed to correctly extract overlapping sub-images. Specialized post-processing procedures could improve the results.

Aside from improving the analysis methods already included in the approach, adding specialized analysis methods for image types, such as line graphs, scatter plots, and photographic images, can further augment the detection capabilities of the approach.

6 CONCLUSION

We introduced an image-based plagiarism detection approach that adapts itself to forms of image similarity found in academic work. The adaptivity of the approach is achieved by including methods that analyze heterogeneous image features, selectively employing analysis methods depending on their suitability for the input image, using a flexible procedure to determine suspicious image similarities, and enabling easy inclusion of additional analysis methods in the future. To derive requirements for our approach, we examined images contained in the VroniPlag collection. This real-world collection is the result of a crowd-sourced project documenting alleged

and confirmed cases of academic plagiarism. From these cases, we introduced a classification of the image similarity types that we observed. We subsequently proposed our adaptive image-based PD approach. Our process integrates perceptual hashing, for which we extended the detection capabilities by including an extraction procedure for sub-images. Since textual labels are common in academic images, we devised and integrated two approaches using OCR to extract text from images and use the textual features for similarity assessments. To address the problem of data reuse, we integrated an analysis method capable of identifying equivalent bar charts. To quantify the suspiciousness of identified similarities, we presented an outlier detection process. The evaluation of our PD process demonstrates reliable performance and extends the detection capabilities of existing image-based detection approaches. We provide our code as open source and encourage other developers to extend and adapt our approach.

REFERENCES

[1] Salha Alzahrani, Vasile Palade, Naomie Salim, and Ajith Abraham. 2011. Using Structural Information and Citation Evidence to Detect Significant Plagiarism Cases in Scientific Publications. *JASIST* 63(2) (2011).

[2] Salha M. Alzahrani, Naomie Salim, and Ajith Abraham. 2012. Understanding Plagiarism Linguistic Patterns, Textual Features, and Detection Methods. In *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, Vol. 42.

[3] Yaniv Bernstein and Justin Zobel. 2004. A Scalable System for Identifying Co-derivative Documents. In *Proc. SPIRE*. LNCS, Vol. 3246. Springer.

[4] Teddi Fishman. 2009. "We know it when we see it"? is not good enough: toward a standard definition of plagiarism that transcends theft, fraud, and copyright. In *Proc. Asia Pacific Conf. on Educational Integrity.*

[5] Bela Gipp. 2014. *Citation-based Plagiarism Detection - Detecting Disguised and Cross-language Plagiarism using Citation Pattern Analysis.* Springer.

[6] Cristian Grozea and Marius Popescu. 2011. The Encoplot Similarity Measure for Automatic Detection of Plagiarism. In *Proc. PAN WS at CLEF.*

[7] Azhar Hadmi, William Puech, Brahim Ait Es Said, and Abdellah Ait Ouahman. 2012. *Watermarking*. Vol. 2. InTech, Chapter Perceptual Image Hashing.

[8] Petr Hurtik and Petra Hodakova. 2015. FTIP: A tool for an image plagiarism detection. In *Proc. SoCPaR.*

[9] Marcin Iwanowski, Arkadiusz Cacko, and Grzegorz Sarwas. 2016. Comparing Images for Document Plagiarism Detection. In *Proc. ICCVG.*

[10] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. 2014. Caffe: Convolutional Architecture for Fast Feature Embedding. In *Proc. Multimedia.*

[11] H.F. Judson. 2004. *The Great Betrayal: Fraud in Science.* Harcourt.

[12] Jan Kasprzak and Michal Brandejs. 2010. Improving the Reliability of the Plagiarism Detection System. In *Proc. PAN WS at CLEF.*

[13] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. ImageNet Classification with Deep Convolutional Neural Networks. In *Proc. NIPS.*

[14] Donald L. McCabe. 2005. Cheating among College and University Students: A North American Perspective. *IJEI* 1, 1 (2005).

[15] Norman Meuschke and Bela Gipp. 2013. State-of-the-art in detecting academic plagiarism. *IJEI* 9, 1 (2013).

[16] Norman Meuschke and Bela Gipp. 2014. Reducing Computational Effort for Plagiarism Detection by using Citation Characteristics to Limit Retrieval Space. In *Proc. JCDL.*

[17] Norman Meuschke, Moritz Schubotz, Felix Hamborg, Tomas Skopal, and Bela Gipp. 2017. Analyzing Mathematical Content to Detect Academic Plagiarism. In *Proc. CIKM.*

[18] Norman Meuschke, Nicolas Siebeck, Moritz Schubotz, and Bela Gipp. 2017. Analyzing Semantic Concept Patterns to Detect Academic Plagiarism. In *Proc. Int. WS on Mining Scientific Publications at JCDL.*

[19] Chirag Patel, Atul Patel, and Dharmendra Patel. 2012. Optical Character Recognition by Open source OCR Tool Tesseract: A Case Study. *Int. J. of Computer Applications* 55, 10 (2012).

[20] Solange de L. Pertile, Viviane P. Moreira, and Paolo Rosso. 2016. Comparing and combining Content- and Citation-based approaches for plagiarism detection. *JASIST* 67, 10 (2016).

[21] Martin Potthast, Benno Stein, Alberto Barrón Cedeño, and Paolo Rosso. 2010. An Evaluation Framework for Plagiarism Detection. In *Proc. COLING.*

[22] Ray Smith. 2007. An Overview of the Tesseract OCR Engine. In *Proc. ICDAR.*

[23] Siddharth Srivastava, Prerana Mukherjee, and Brejesh Lall. 2015. imPlag: Detecting image plagiarism using hierarchical near duplicate retrieval. In *Proc. INDICON.*

[24] Benno Stein, Sven Meyer zu Eissen, and Martin Potthast. 2007. Strategies for Retrieving Plagiarized Documents. In *Proc. SIGIR.*

[25] Satoshi Suzuki and Keiichi Abe. 1985. Topological Structural Analysis of Digitized Binary Images by Border Following. *CVGIP* 30, 1 (1985).

[26] K. Vani and Deepa Gupta. 2016. Study on Extrinsic Text Plagiarism Detection Techniques and Tools. *J. Engin. Sc. & Techn. Review* 9, 5 (2016).

[27] Debora Weber-Wulff. 2014. *False Feathers: A Perspective on Academic Plagiarism.* Springer.

A TEST CASES

Figures 5-19 show the test cases of our study. Each case shows the source image on the left and the reused image on the right. Due to space limitations, we only cite the identifier of the investigation in the VroniPlag collection. The identifier consists of the case id, e.g., Ry, followed by the page number where the fragment appears in the later document. Appending the identifier to the base URL de.vroniplag.wikia.com/wiki/ shows each case and allows identifying its source http://de.vroniplag.wikia.com/wiki/Ry/073.

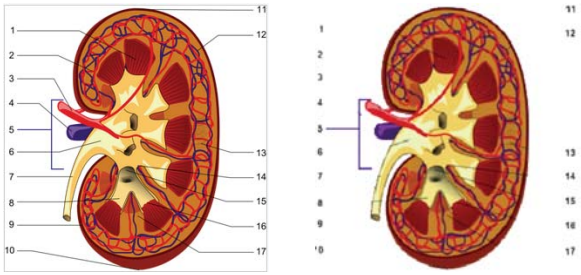

Figure 5: Case 1 - near copy illustration [Dsa/014].

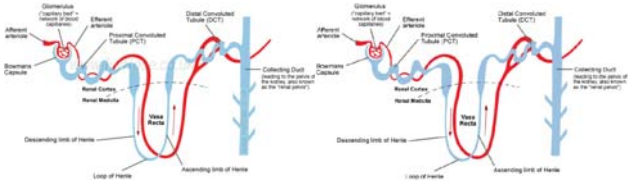

Figure 6: Case 2 - near copy illustration [Dsa/015].

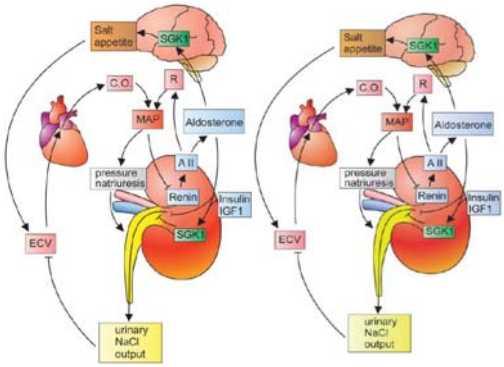

Figure 7: Case 3 - near copy illustration [Dsa/025].

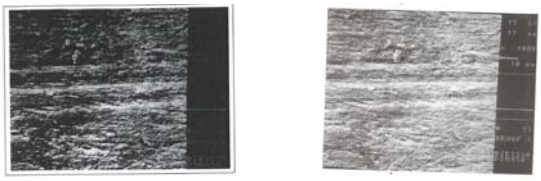

Figure 8: Case 4 - near copy microscope image [Cz/048].

Vertikale Auflösung	5 Å
Horizontale Auflösung	500 Å
Digitale Rauschunterdrückung	möglich
Meßbereich vertikaler Strukturen	100 Å - 655.000 Å

Vertikale Auflösung	5 Å
Horizontale Auflösung	500 Å
Digitale Rauschunterdrückung	möglich
Meßbereich vertikaler Strukturen	100 Å - 655.000 Å

Figure 9: Case 5 - near copy table [Cz/039].

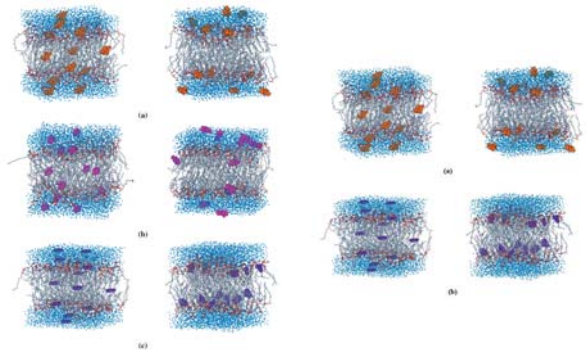

Figure 10: Case 6 - weakly altered illustration [Ry/073].

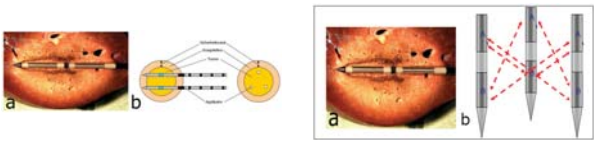

Figure 11: Case 7 - weakly altered illustration [Chh/005].

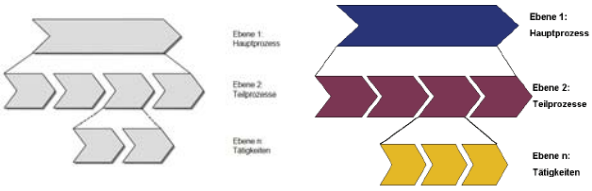

Figure 12: Case 8 - moderately altered figure [Ab/017].

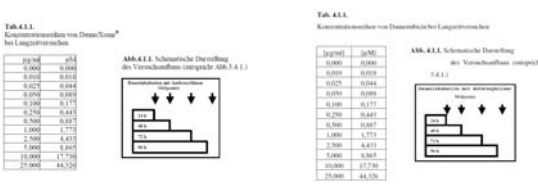

Figure 13: Case 9 - moderately altered table [Jus/029].

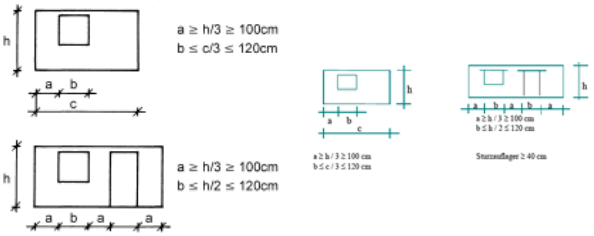

Figure 14: Case 10 - strongly altered sketches [Aos/193].

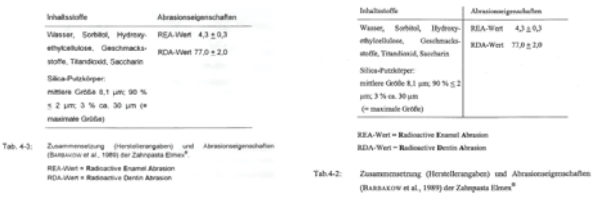

Figure 15: Case 11 - strongly altered table [Cz/035].

Zelllinie	Verdopplungszeit in der exponentiellen Wachstumsphase
Cado ES-1	ca. 32 Stunden
STA-ET-1	ca. 48 Stunden
STA-ET-2.1	ca. 76 Stunden
VH-64	ca. 20 Stunden

Zelllinie	Verdopplungszeit in der exponentiellen Wachstumsphase
Cado ES-1	ca. 32 Stunden
STA-ET-1	ca. 48 Stunden
STA-ET-2.1	ca. 76 Stunden
VH-64	ca. 20 Stunden

Figure 16: Case 12 - strongly altered table [Jus/022].

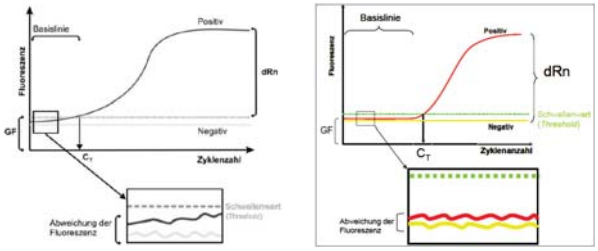

Figure 17: Case 13 - strongly altered line chart [Ad/068].

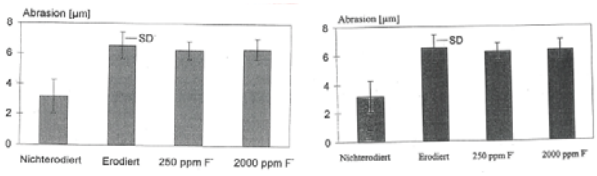

Figure 18: Case 14 - near copy bar chart [Cz/047].

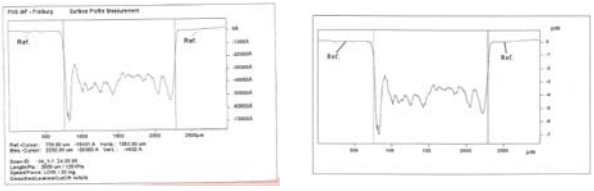

Figure 19: Case 15 - near copy line chart [Cz/044].