Automated Processing of scanned historic watermarks

A Comparison of Feature Extraction Techniques for Binarized Content-Based Image Retrieval

Sydney Kho[1]

Supervisors: Dr. Martin Skrodzki[1]**, Dr. Jorge Martinez Castaneda**[1]

[1]**EEMCS, Delft University of Technology, The Netherlands**

A Thesis Submitted to EEMCS Faculty Delft University of Technology,
In Partial Fulfilment of the Requirements
For the Bachelor of Computer Science and Engineering
June 21, 2024

Name of the student: Sydney Kho
Final project course: CSE3000 Research Project
Thesis committee: Dr. Martin Skrodzki, Dr. Jorge Martinez Castaneda, Dr. Christoph Lofi

Abstract

Feature extraction techniques for content-based image retrieval are explored, focusing on black-and-white images in the context of historical watermarks. Orthogonal moments and texture features are found to be most applicable. Seven methods are evaluated: four different orthogonal moments, Gabor features, and two novel combinations of orthogonal moments with Gabor features. Retrieval effectiveness is judged based on the precision-recall curve and mean average precision, and watermarks are considered when unchanged, rotated, sheared and both rotated and sheared. The results demonstrate that research into improving efficient grayscale image representation does not translate over to improvements with black-and-white images. As it stands, the basic Zernike moments and novel Gabor-Zernike features are most effective.

1 Introduction

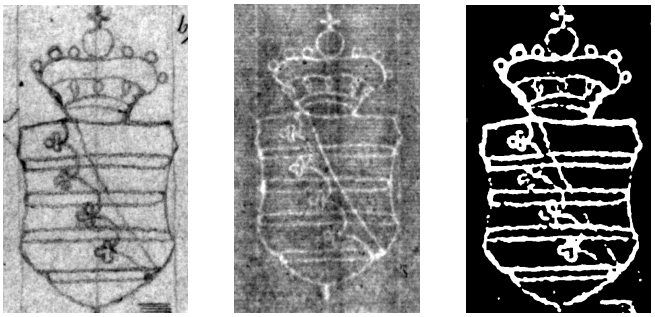
In this work, the term watermarks refers to images embedded in paper, created when drying the paper during manufacturing and used to identify the manufacturer. This information can help historians relate the time and place different works were produced, which could be invaluable for finding out more information about a historical document. Unfortunately, this requires someone to search through archives of up to thousands of watermarks due to a general lack of digitized datasets and a lack of generalized tools. One system that does exist for organizing watermarks is the Bernstein Project[1], which provides tools to manually improve the clarity of, and organize watermarks. This system does not, however, automatically organize the images.

In previous work a prototype system has been created to automate the organization of watermarks [1]. It works on both images of the watermark traced out by a person using pencil (as in Figure 1a), and on images of the original watermark in the paper (as in Figure 1b). These are referred to as traced and untraced, respectively, and an example of each can be seen below in Figure 1. The system in [1] first clarifies the watermarks by increasing the contrast through various techniques, before applying a threshold to decide which pixels are, and are not, part of the watermark. The result is a black and white image containing only the watermark, as in Figure 1c. This maximizes the effectiveness of the final step: extracting features from the image to make fast, automatic comparison easy. The chosen features in [1] were effective, but not necessarily optimal as they were chosen through trial and error. It is therefore necessary to find image features that are quantitatively best at retrieving binarized watermarks.

The main question to be answered is as follows: **"What image features are the most effective for retrieving similar, binarized watermarks from a set of historical documents?"**. This is further divided into four sub-questions:

1. What types of features are relevant for recognition of binarized watermarks?

[1] https://memoryofpaper.eu/



(a) (b) (c)

Figure 1: (a) A traced watermark, (b) An untraced watermark, (c) A binarized watermark

2. Which specific image features may perform well, and how do they actually perform at watermark retrieval?
3. What impact do the parameters of the techniques have, and how can they be optimized?
4. How can different techniques be combined to further improve performance of the system?

For the first question, several categories of feature extraction are identified and considered based on a survey by Qi et al. [2] and another by Mikolajczyk et al. [3]. Based on this, multiple specific techniques are chosen and evaluated to answer the second question. As a part of the evaluation, parameters are optimized for the best performance on binarized watermarks. Finally, an attempt is made at combining the most effective features to further improve performance.

2 Related Works

In 1962, Hu introduced the concept of moment invariants, which are one of the first widely used image features that can be used to represent and compare images, even if they are different in scale, rotation or translation [4]. Since then, many other techniques for representing images for effective comparison have been developed: orthogonal moments efficiently store information with no redundancy while being easily comparable [2]. local descriptors such as the Scale Invariant Feature Transform (SIFT) stores highly invariant descriptions of interest points in an image [5], and the Generic Fourier Descriptor (GFD) represents the images in the frequency domain to be robust to noise [6]. What they all have in common is that they were developed for color or grayscale images, rarely considering binarized images as its own category. Therefore, there is a need to evaluate these image features specifically for binarized images.

Feature extraction techniques can broadly be divided into five categories [2; 3]. In the following subsections, each one is described and their applicability to binarized watermarks is considered.

2.1 Local Descriptors

Features based on local descriptors are perhaps the most intuitive. Regions of interest are identified and descriptors are

generated for each. This mimics how a human may remember specific interesting parts in an image. Specific techniques vary in how regions of interest are found and what the actual descriptors look like. For example, SIFT uses the Laplacian-of-Gaussians to find regions of interest, and describes them using a histogram of gradient orientations [7].

Local descriptors vary in the time needed to compute them depending on how complex the image is. For images containing few objects, such as watermarks, they are quite fast to compute. They are also very flexible, as individual descriptors can be anywhere in the image. This makes them highly invariant, including to perspective changes.

The main disadvantage comes from the need to compare images. Not only must every image be compared to every other image, but for each descriptor, an attempt must be made to match it to another descriptor. This process can be made somewhat faster using various approximate nearest neighbor algorithms, such as the hierarchical k-means algorithm [8], or the randomized kd-tree algorithm [9], but this does not avoid the need to match many keypoints, rather than comparing a single vector for each image. Additionally, there are issues with polysemy, where the same object may be represented by many different descriptors, and synonymy, where one descriptor can be applied to many (semantically different) objects [10]. This leads to both over- and under-representation of visual patterns. Local descriptors may not be too effective for representing watermarks specifically, given that many watermarks have common components. For example, many watermarks contain shields, but they are not necessarily related. An example of this is in Figure 2.

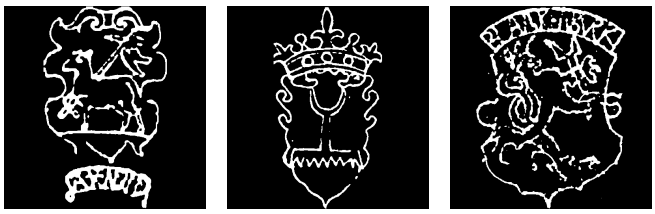


Figure 2: Examples of various decorated shields from the training set (explained in Section 4.2), which should not be matched to each other."

2.2 Texture Features

Texture features do not search for specific regions of interest, but instead quantify the texture of the image in its entirety. For example, Gabor filters can be used to isolate frequency and orientation information from the image, which can be used as a feature vector. In the case of [11], the mean and standard deviation of each filtered image is used. Another method is the local binary pattern. This considers a neighborhood around each pixel, and counts the number of occurrences of particular patterns. The resulting histogram can be used as a feature vector.

These features process the entire image and use measures that are relatively simple. While it's possible to make them invariant to rotation, this results in simplistic features that are even less discriminative. For example, it was already possible for two images to result in the same local binary pattern histogram by having the same patterns but in different locations in the image. If different orientations of the same pattern are considered identical, this would allow for even more images that result in the same histogram. In practice, the input watermarks are mostly normalized with regard to rotation, so texture features may still be fairly effective.

2.3 Image Moments and Moment Invariants

Image moments are statistical measures of the shape of an object in an image. Moments can be used to derive features that are invariant to affine transformations, as with the Hu moment invariants [4]. One commonly used type of image moment and one of the first proposed, are the Zernike moments. They are based on the Zernike polynomials, which are a set of orthogonal moments. The image moments are then calculated by projecting the image onto different polynomials. [12]

The Zernike moments are rotation invariant, robust to noise and have been shown to be very effective for content-based image retrieval [13]. The main disadvantage to using them is that higher order moments quickly become very intensive to compute. This can be mitigated by using the recurrence relation between moments, but this introduces another problem. Computers introduce small numerical imprecisions, which compound because higher order moments are dependent on lower order ones. On the other hand, very high order moments are rarely needed, as good image reconstruction can be done with as low as 16th order moments [12]. This would suggest most information about an image can be captured in relatively few moments.

2.4 Frequency Transforms

Frequency transforms can be used to analyze the image in the frequency domain, rather than the spatial domain. The main benefit is that this is highly resistant to noise. One example is the Generic Fourier Descriptor (GFD) [6], which first manipulates the image to be rotation invariant after the transform. Then the polar Fourier transform is used to obtain magnitudes for each frequency, up to a high enough frequency to perform well.

The GFD has been shown to be a very effective way to compare images. However, for the application to binarized images the Fourier transform has to represent very strong edges, which may require high frequency information. This makes it inefficient at storing the images.

2.5 Dimensionality Reduction

Dimensionality reduction is the process of reducing the number of dimensions in high-dimensional data, while keeping relations intact. Many such techniques exist, like Principal Component Analysis (PCA), Singular Value Decomposition (SVD) and Locally Linear Embedding (LLE) [14; 15; 16]. These can be directly applied by treating the images themselves as vectors, or used together with a different technique to reduce the amount of space it takes up.

Applying dimensionality reduction directly to the images themselves would require either considering the entire dataset at once, or some subset for training. This is because PCA

and LLE both calculate some transformation from the original feature space to a lower-dimensional sub-space by using the relationships between all input vectors. It's possible to use less data to decide on a transformation, but then an optimal outcome is not guaranteed. More logical would be to use dimensionality reduction on high-dimensional data from other techniques, but other techniques for image feature extraction don't typically result in very high-dimensional data. Additionally, there is no control over what exactly the reduction in dimensionality preserves: a rotated image may result in a completely different low-dimensional representation from its original counterpart. Overall, dimensionality reduction is not very applicable to the use case of feature extraction for binarized watermarks.

2.6 Summary of Relevant Works

Image moments seem to be the best approach for representing binarized watermarks, as they provide highly compact, invariant representations. Texture features are also promising, especially given that the watermarks are easy to normalize in terms of scaling and translation. While frequency transforms show potential, they do not fit well with binarized images, as very high frequency information is needed to represent the strong edges. Local descriptors are flexible, but make quick comparison between watermarks challenging, and dimensionality reduction techniques do not align well with the requirements of watermark representation.

Therefore, this paper will focus on investigating specific different image moments and texture features, focusing on their effectiveness for the retrieval of binarized historical watermarks.

3 Overview of Image Moments

Content-based image retrieval (CBIR) is a technique that allows one to search and retrieve images from large databases based on the content they contain rather than textual metadata. This approach is particularly valuable for historical documents, where textual metadata may be limited or nonexistent.

3.1 Image Moments

Image moments are a set of numbers calculated from the image itself, and they can describe various properties of the image such as area, centroid, and orientation. The concept of moments in image analysis is analogous to moments in physics, where they describe the distribution of mass in a physical object. In the context of images, moments provide features that can be invariant to translation, rotation, and scaling.

Moments can be computed in different coordinate systems, with Cartesian and polar coordinates being the most common. Cartesian moments are based on the traditional x and y coordinates of the image, while polar moments are computed in terms of radial distance and angle, which can isolate the rotation of the image to obtain rotation invariance.

Mathematically, the image moment is the projection of the image function f onto a subspace made up of a set of basis functions $\{V_{nm} : (n, m) \in \mathbb{Z}\}$ [2]:

$$\langle f, V_{nm} \rangle = \iint_D V_{nm}^* f(x, y) dx dy$$

Where $*$ is the complex conjugate and D is the domain. The choice in V_{nm} is what differentiates image moments.

3.2 Orthogonality in Moments

Orthogonality is a key property in the context of moments, referring to the mathematical independence of basis functions used to compute the moments. When moments are orthogonal, each moment captures unique information about the image, minimizing redundancy and increasing efficiency. For instance, Zernike moments are orthogonal and thus provide a more compact and discriminative representation of image features compared to non-orthogonal moments. A set of basis functions is considered orthogonal when for all basis functions $V_{nm}, V_{n'm'}$:

$$\langle V_{nm}, V_{n'm'} \rangle = \delta_{nn'} \delta_{mm'}$$

where δ_{ij} is the Kronecker delta function, defined as:

$$\delta_{ij} = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}$$

This means the inner product between any two different basis functions is zero, and the inner product between a basis function and itself is one.

3.3 Orthogonal Moments in Polar Coordinates

Polar moments are computed using a set of orthogonal basis functions defined in polar coordinates. This involves parameterizing the moments in terms of radial distance r and angular aspect θ, such that the basis set can be decomposed into a set of radial basis functions $R_n(r)$ and a set of angular basis functions $A_m(\theta)$.

$$V_{nm}(rcos(\theta), rsin(\theta)) \equiv V_{nm}(r, \theta)$$
$$= R_n(r) A_m(\theta)$$

Where $A_m(\theta) = e^{im\theta}$ and $R_n(r)$ is some orthogonal function. This isolates any rotation to the phase of the image moment, leaving the magnitude unchanged and therefore providing rotation invariance. At the same time, this means the Cartesian coordinates typically used in digital images have to be converted to polar coordinates, leading to small rounding errors.

3.4 Orthogonal Moments in Cartesian Coordinates

In Cartesian coordinates, the moments are either calculated by mapping the range of pixels to a smaller range such as $[0, 1]$ or $[-1, 1]$ depending on the exact basis set used. The general form is as follows:

$$V_{nm} = P_n(x) P_m(y)$$

where $P_n(x)$ and $P_m(y)$ are chosen polynomials of order n and m respectively. For example, the Legendre moments use the Legendre polynomials [12]. Important to note is that these are not rotation invariant.

4 Methodology

The goal is to develop a CBIR system for binarized images and evaluate the effectiveness of various feature extraction techniques. Specifically focusing on orthogonal image moments and Gabor features. These will be compared in terms of precision, recall and mean average precision.

4.1 Considered Features

An outline is given of why each feature was chosen. For the polar moments, the image is inscribed with a circle, and only the pixels in this circle are considered.

For the Gabor features, the OpenCV function `cv2.getGaborKernel()` was used. The implementation of the Zernike moments from the Python package mahotas version 1.4.14 was used, specifically `mahotas.features.zernike_moments()`. For the Bessel-Fourier moments, scipy version 1.13.0 was used. The specific functions are `scipy.special.jn_zeros()` and `scipy.special.jv()`. The Bessel-Fourier, Legendre and Tchebichef moments were implemented by the author, and tested by comparing the generated polynomials to their definitions, as well as reconstructing the original image from the moments.

Zernike Moments

Zernike moments were chosen because they are commonly used polar moments. They are one of the first suggested orthogonal moments by Teague [12].

Bessel-Fourier Moments

Bessel-Fourier moments are a more recent set of polar moments. They are intended to be an improvement on the Zernike moments, as well as other polar moments that came after, such as the Fourier-Mellin moments [17].

Legendre Moments

Legendre moments were chosen because they are well known discrete Cartesian moments. They were first suggested in 1980 along with the Zernike moments in [12].

Tchebichef Moments

Tchebichef moments are another set of discrete Cartesian moments. They were developed after the Legendre moments and are supposed to be an improvement on them [18].

Gabor Features

As a baseline, simple Gabor features based on the mean and standard deviation of Gabor filtered images are included. To be noted is that this is not the same algorithm as [11], as it does not include correlation information. The reason for this is that it will make it easier to compare the more complex moments with simple statistical measures.

Gabor-Zernike and Gabor-Legendre Features

To improve the Gabor features, two novel techniques are proposed. Rather than taking the global mean and standard deviation, more complex moments of the Gabor filtered images can be taken. This combines the ability to isolate frequency and rotation information with the very efficient image representation that moments provide. The choice was made to combine Gabor filtering with the two image moments first suggested by Teague [12] to evaluate how effective the concept is. The new techniques first filter the image at 4 orientations and 4 scales, similar to [11]. Then the Zernike or Legendre moments of the filtered images are calculated, and these are appended to form a single feature vector.

4.2 Dataset

(a) (b) (c)

Figure 3: (a) Original image, (b) Image after pipeline (black borders cropped out), (c) Image after manual editing and normalization

A dataset was provided by the German Museum of Books and Writing[2]. This consisted of many pictures taken of watermarks, both traced and untraced. An example of a traced watermark can be seen in Figure 3a. The dataset was organized into folders containing related watermarks, so four of these folders were arbitrarily chosen to be processed. To make it easier to process the images, and to ensure the images resulted in a meaningful evaluation, they were chosen based on four criteria:

1. There must be at least three representations of the same watermark.
2. The watermark cannot be deteriorated too much. Specifically: at most one fourth of the watermark may be missing.
3. The image must contain exactly one watermark
4. The watermark must be traced

At least three copies should be present so that when a ranking is created, there are at least two images that are expected to be high up. Similarly, criterion (2) was not only chosen to make processing easier, but a watermark that is too deteriorated is not expected to be matched to its complete counterparts. Criteria (3) and (4) were only to ease the process of binarizing the watermarks.

After obtaining the images to process, they were binarized using the (automatic) harmonization algorithm for traced images from [1], resulting in images like Figure 3b. This algorithm is not perfect, so watermarks were edited manually by the author to add missing parts or to remove parts that were included by mistake. Care was taken to only follow the tracings present on paper, and not to complete watermarks with missing parts. To further normalize the images, empty borders were cropped, and the images were resized such that the largest dimension of the watermark was 512 pixels wide. Then the image was padded in the other dimension to make the image 512x512 pixels. Although resizing the original images does result in some loss of information, this was not a

[2] www.dnb.de

concern because the goal is to find similar images. The small changes are unlikely to change the results in any major way, ensuring similar images stay comparable. Additionally, having every image be the same size was important for a consistent evaluation. An example of the final result can be seen in Figure 3c.

In total, this resulted in 311 images which were split 80/20 into training and evaluation, however it was later found that the training set had particularly large groups of one type of image. There were two large groups of thirty eagle watermarks, as well as some smaller ones, making up 80 of the 246 images. Some examples of these are in Figure 4. To reduce the over-representation of the eagle watermarks, only 3 images of each group were included. In the end, 65 images were used for evaluation, and 81 images were used for training. This means the actual split was 55/45.

(a) (b) (c)

Figure 4: Eagles in different groups, having: (a) 31 representations, (b) 29 representations, (c) 10 representations

It's possible that input images may be distorted. A watermark could be rotated, or the image may not be taken from directly above the watermark. To evaluate both cases, three variations of the dataset were synthetically created. The first is rotated randomly, the second is sheared in random directions by random amounts such that the shear angle is at most 30 degrees, and the final set has both operations done to it.

4.3 Evaluation

Features are extracted from all images. For each image, a ranking is produced by using the Euclidean distance between the dataset images and the input image. The rankings are evaluated using the precision and recall when retrieving k images, $P(k)$ and $R(k)$ respectively. They are commonly used when evaluating information retrieval tasks. These are defined to be [19]:

$$P(k) = \frac{\text{Relevant retrieved images}}{k}$$
$$R(k) = \frac{\text{Relevant retrieved images}}{\text{All relevant images}}$$

These metrics are used to produce the precision-recall curve, which shows how the proportion of images that are relevant changes as more relevant images are retrieved. It illustrates the trade-off between maximizing the number of relevant images retrieved and ensuring that most of the retrieved images are indeed relevant. Additionally, the mean Average Precision (mAP) is used to give an idea of the overall performance:

$$\text{AP} = \sum_{k=1}^{n} P(k)\Delta R(k)$$
$$\text{mAP} = \frac{1}{I}\sum_{i=1}^{I} AP(i)$$

Where n is the total number of images retrieved, $\Delta R(k)$ is the change in recall from retrieving $k-1$ to k images and I is the number of images in the dataset. AP is the average precision, defined to be the area under the precision-recall curve [19]. The mAP is then defined to be the mean average precision over all rankings that are done. This provides a way to characterize the overall retrieval performance.

4.4 Choice of Parameters

	Maximum moment order				
	6	8	10	12	14
Zernike	0.630	0.641	**0.697**	0.679	0.672
Bessel-Fourier	0.646	**0.659**	0.641	0.620	0.592
Legendre	0.670	**0.764**	0.745	0.708	0.684
Tchebichef	0.630	0.706	0.742	**0.764**	0.756
Gabor-Zernike	0.762	**0.772**	0.754	0.766	0.765
Gabor-Legendre	0.766	**0.818**	0.803	0.770	0.765

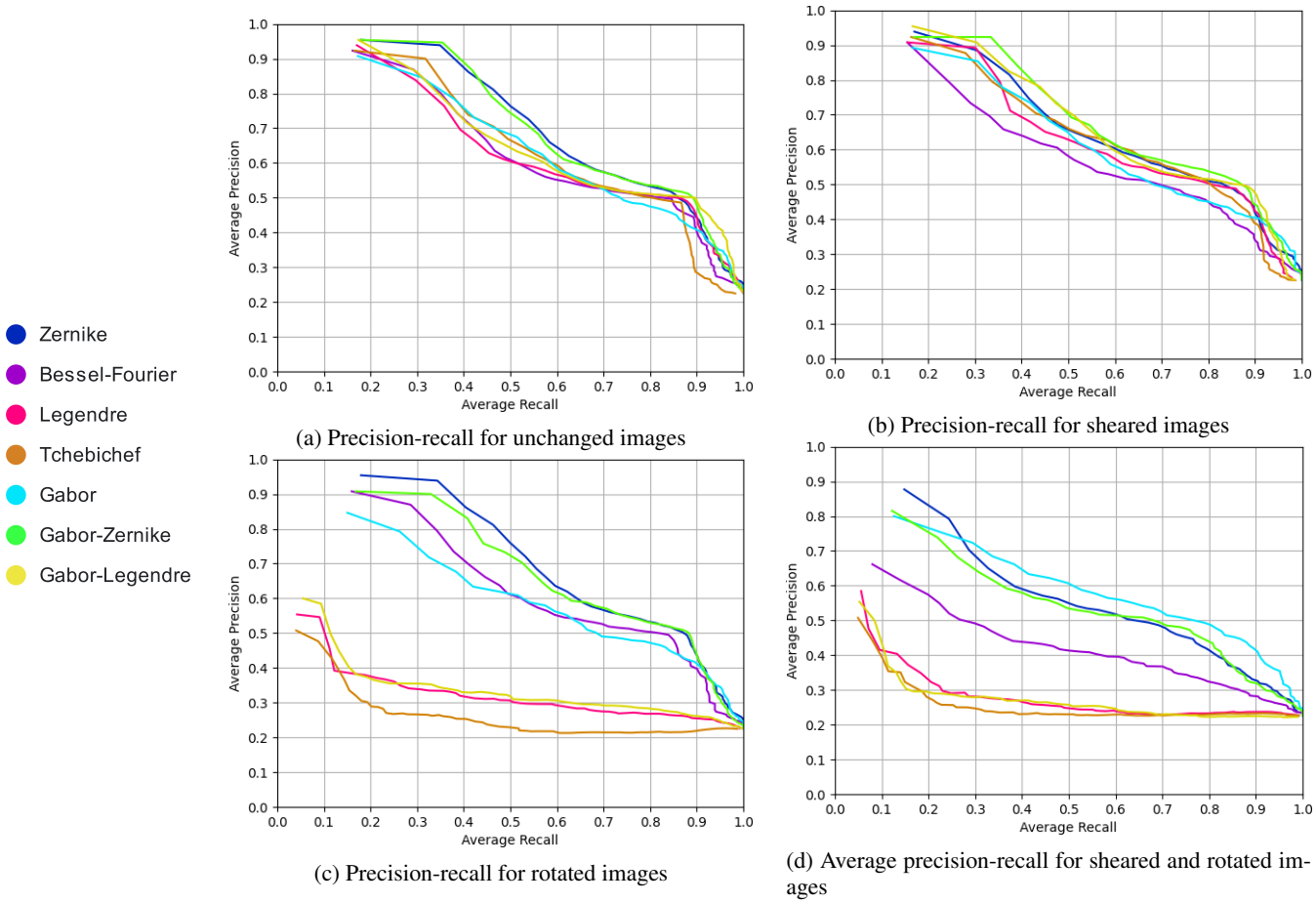
Table 1: An overview of the mean average precision for each moment-based technique where the maximum order ranges from 6 to 14. In bold are the highest values for each technique, corresponding to the maximum moment order used.

To decide the number of moments for each technique, a preliminary experiment was done on the training data. This consisted of 27 groups of 3 images each, with 81 images in total. This is to ensure no particular watermark was over-represented. For each technique, the evaluation was run with moments of order 2 up to 16. Whichever moment order resulted in the highest mAP was chosen. The maximum moments order represents the highest amount of detail each particular feature can capture. With too few moments, the image won't be described sufficiently to discriminate between them. With too many, unnecessary detail may be captured which would effectively be noise. The results are in Table 1. Scores not included were all lower.

5 Results

Based on the mAP, each technique performs best when the highest order moments are of order eight to twelve. For higher order moments, performance starts to fall. This could be because they are not being calculated accurately. Another possibility is that higher order moments simply capture such high detail that they are effectively capturing noise.

The different precision-recall curves for unchanged, rotated, sheared and both rotated and sheared images are in Figure 5. The mAP for each technique under different conditions is in Table 2.



(a) Precision-recall for unchanged images

(b) Precision-recall for sheared images

(c) Precision-recall for rotated images

(d) Average precision-recall for sheared and rotated images

Figure 5: Precision-recall curves for each technique when the images are unchanged, sheared, rotated, and both sheared and rotated. Precision and recall both range from 0 to 1 and are dimensionless.

With the basic, unchanged images the precision changes very similarly with recall for all techniques. When about one third of the images are correctly recalled, the precision is very high. Then the precision falls to about 50% at 90% recall. For the final 10% the precision drops quickly as each technique has trouble finding the more difficult images. Zernike moments and Gabor-Zernike features both seem to have generally higher precision than the others, which is supported by their mAP being the largest too.

For rotated images the results are very similar for the rotation invariant moments. The other techniques perform generally badly, though this is expected. Notably, Gabor-Zernike features now have a lower mAP than Zernike moments. Gabor manages to not do much worse, likely because the overall texture of the image remains similar.

Shearing also has very similar results to the unchanged images, with the mAP of most techniques being the same or slightly lower. The only exception to this are the Gabor-Legendre features, which have a slightly higher mAP. Both the Gabor-Zernike features and Gabor-Legendre features now have a higher mAP than the Zernike moments. When both shearing and rotation is applied, the rotation invariant moments once again come out on top, with Zernike having the highest mAP. Simple Gabor features perform the best, likely because they are the most invariant, making use of only the mean and standard deviation of the image.

The combined techniques were effective and got similar results to their counterparts in all categories. Their expected benefit was to capture the texture more effectively than the basic Gabor features. This does seem to be the case, as When the image is sheared or rotated, Gabor-Zernike features have

	Mean Average Precision			
	Base	Rot.	Shear	Both
Zernike	0.881	0.876	0.819	0.693
Bessel-Fourier	0.797	0.792	0.738	0.531
Legendre	0.802	0.393	0.802	0.352
Tchebichef	0.821	0.335	0.804	0.333
Gabor	0.819	0.759	0.798	0.739
Gabor-Zernike	0.881	0.856	0.855	0.663
Gabor-Legendre	0.821	0.407	0.834	0.344

Table 2: Mean average precision for each technique when the images are unchanged, rotated, sheared and both rotated and sheared.

higher mAP than the Gabor features. Only once the image is both sheared and rotated, does Gabor do better. Gabor-Legendre moments mostly seem to be dragged down by their lack of rotation invariance, mostly having lower mAP than the Gabor features and being almost the same as the Legendre moments.

Tchebichef and Bessel-Fourier don't seem to improve on Legendre and Zernike respectively when applied to binarized images. The newer moments' curves and mAP are very similar to their counterparts, and generally worse. It's likely these new moments take advantage of the additional detail that grayscale provide, but is unavailable for binarized images.

Overall, both Zernike moments and Gabor-Zernike features seem reasonable choices. On the unchanged images they perform the same. Gabor-Zernike features are slightly worse with rotated images, but better with sheared images. When images are expected to vary in both rotation and shearing, simple Gabor features are actually best, presumably due to their generally high invariance.

6 Responsible Research

The dataset used for this paper was provided by the German Museum of Books and Writing, containing many historical watermarks. These images do not contain any personal or sensitive information. Unfortunately, the dataset itself is not publicly available. Those who are interested in accessing the dataset can request access directly from the museum, but there is no guarantee it will be provided.

To address potential bias, specifically the overrepresentation of eagle watermarks, a subset of the training images was used. The first three images were included from each group. The resulting dataset better represents the variability of watermarks more broadly, making the results more generalizable.

Throughout the research process, every step was documented and included in this paper. The code for automatically binarizing images is publicly available [1], as well as the code to generate the variations on the dataset (rotated, sheared, and both), each feature extraction technique and the evaluation. All of these are available at the TU Delft Repositories[3]. Unfortunately, fixing the binarized images was a manual process, so this can not be replicated perfectly.

In the writing of this paper, ChatGPT[4] was used. It was used only to help structure the text, and no content was directly taken from it. It assisted in outlining some sections in the form of bullet points, as well as brainstorming information to include. An example prompt (assuming the model has been provided context) would be: "Give an outline in bullet points for the methodology section. It should at least include which image features were considered, how the dataset was generated and how the evaluation was done."

7 Conclusions and Future Work

In this paper, multiple image features were considered, implemented and tested for the purpose of retrieving binarized watermarks. It was found that image moments and texture features were the most applicable for this use case. Older and newer orthogonal moments were tested, showing that newer image moments do not result in improvement for binarized images. Gabor texture features, which mimic the way cells in the mammalian visual cortex function, were tested and combined with image moments, which improved their recall and invariance to shearing and rotation. After testing and evaluating their precision and recall, it was found that both Zernike moments and Gabor-Zernike features perform similarly, and are both good choices for watermark retrieval. Gabor-Zernike features are slightly better under shearing, while Zernike moments are better under rotation and when shearing and rotation are combined.

First, the parameters for the Gabor filters were based on [11]. It's possible other parameters could still have improved performance. Improvements in image moments for grayscale images do not seem to correspond to improvements for binarized images. More research is needed into moments that capture binarized images more effectively. It was also found that the maximum order of moments to most effectively retrieve watermarks was around eight to twelve. It was unclear if this was because of numerical imprecision, because higher moments capture unnecessary detail, or possibly something else.

[3] https://repository.tudelft.nl

[4] https://chatgpt.com

References

[1] D. Banță, A. Lantink, V. Petkov, A. Marin, and S. Kho, "A watermark recognition system: An approach to matching similar watermarks," Delft University of Technology, Tech. Rep., 2023. [Online]. Available: http://resolver.tudelft.nl/uuid:e8dfbd63-ae54-4159-b786-d1d8c64dc827

[2] S. Qi, Y. Zhang, C. Wang, J. Zhou, and X. Cao, "A survey of orthogonal moments for image representation: Theory, implementation, and evaluation," *ACM Computing Surveys*, vol. 55, no. 1, p. 1–35, Nov 2021.

[3] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 10, pp. 1615–1630, 2005.

[4] M.-K. Hu, "Visual pattern recognition by moment invariants," *IRE Transactions on Information Theory*, vol. 8, no. 2, pp. 179–187, 1962.

[5] D. Lowe, "Object recognition from local scale-invariant features," in *Proceedings of the Seventh IEEE International Conference on Computer Vision*, vol. 2, 1999, pp. 1150–1157 vol.2.

[6] D. Zhang and G. Lu, "Shape-based image retrieval using generic fourier descriptor," *Signal Processing: Image Communication*, vol. 17, no. 10, p. 825–848, Nov 2002.

[7] W. Burger and M. J. Burge, *Scale-Invariant Feature Transform (SIFT)*. London: Springer London, 2016, pp. 609–664. [Online]. Available: https://doi.org/10.1007/978-1-4471-6684-9_25

[8] M. Muja and D. G. Lowe, "Fast approximate nearest neighbors with automatic algorithm configuration," in *International Conference on Computer Vision Theory and Application VISSAPP'09)*. INSTICC Press, 2009, pp. 331–340.

[9] C. Silpa-Anan and R. Hartley, "Optimised kd-trees for fast image descriptor matching," in *2008 IEEE Conference on Computer Vision and Pattern Recognition*, 2008, pp. 1–8.

[10] J. Yuan, Y. Wu, and M. Yang, "Discovery of collocation patterns: from visual words to visual phrases," in *2007 IEEE Conference on Computer Vision and Pattern Recognition*, June 2007, pp. 1–8.

[11] H.-H. Bu, N.-C. Kim, B.-H. Lee, and S.-H. Kim, "Content-based image retrieval using texture features extracted from local energy and local correlation of gabor transformed images," *Journal of Information Processing Systems*, vol. 13, no. 5, p. 1372 – 1381, 2017, cited by: 3; All Open Access, Bronze Open Access. [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85032986260&doi=10.3745%2fJIPS.02.0075&partnerID=40&md5=b217a9626d1da0492ac62ae986ba3024

[12] M. R. Teague, "Image analysis via the general theory of moments∗," *J. Opt. Soc. Am.*, vol. 70, no. 8, pp. 920–930, Aug 1980. [Online]. Available: https://opg.optica.org/abstract.cfm?URI=josa-70-8-920

[13] S. Qi, Y. Zhang, C. Wang, J. Zhou, and X. Cao, "A survey of orthogonal moments for image representation: Theory, implementation, and evaluation," *ACM Comput. Surv.*, vol. 55, no. 1, nov 2021. [Online]. Available: https://doi.org/10.1145/3479428

[14] K. Pearson, "Liii. on lines and planes of closest fit to systems of points in space," *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, vol. 2, no. 11, pp. 559–572, 1901. [Online]. Available: https://doi.org/10.1080/14786440109462720

[15] V. Klema and A. Laub, "The singular value decomposition: Its computation and some applications," *IEEE Transactions on Automatic Control*, vol. 25, no. 2, pp. 164–176, 1980.

[16] L. K. Saul and S. T. Roweis, "An introduction to locally linear embedding," in *Journal of Machine Learning Research*, 2001. [Online]. Available: https://api.semanticscholar.org/CorpusID:72532

[17] B. Xiao, J.-F. Ma, and X. Wang, "Image analysis by bessel–fourier moments," *Pattern Recognition*, vol. 43, no. 8, pp. 2620–2629, 2010. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0031320310001330

[18] R. Mukundan, S. Ong, and P. Lee, "Image analysis by tchebichef moments," *IEEE Transactions on Image Processing*, vol. 10, no. 9, pp. 1357–1364, 2001.

[19] M. Zhu, "Recall, precision and average precision," 09 2004.