



# **Curve Reconstruction and Approximation in Binarised Scanned Historic Watermark Images**

**A Study of Techniques Aiding Binarisation for an Automated Watermark Similarity-matching Pipeline**

**Vladimir Plamenov Petkov<sup>1</sup>**

**Supervisor(s): Dr. Martin Skrodzki<sup>1</sup>, Dr. Jorge Martinez Castaneda<sup>1</sup>**

**<sup>1</sup>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 23, 2024

Name of the student: Vladimir Petkov

Final project course: CSE3000 Research Project

Thesis committee: Dr. Martin Skrodzki, Dr. Jorge Martinez Castaneda, Dr. Christoph Lofi

An electronic version of this thesis is available at <http://repository.tudelft.nl/>.

## Abstract

A curve is a continuously bending line with no angles that can be found anywhere in the real world, forming shapes and outlines. They are also the building blocks of historic watermarks, imprinted images on paper that may be used to identify its manufacturers. Their shapes consist of curves as bent wires are used in their production process. Often, the processing of scans of those curves may introduce gaps or a degraded quality which could be corrected by reconstructing the curves in those gaps. Curve reconstruction is a fundamental problem with many research applications, one of which is the reconstruction of curves for binarised scans of historic watermarks. In this paper, a data generation approach is proposed for the simulation of the watermark curves domain through singular automatically generated curves and human-drawn sketches which are then used along binarised watermark scans. I propose a hybrid method combining machine-learning and analytical approaches for curve reconstruction, aiming to leverage their advantages together. The method is compared to its components separately. Quantitative results against them demonstrate the superiority of the pure machine learning approach, as well as the need for more research into potentially better analytical components and a more realistic domain simulation.

## 1 Introduction

Historic watermarks are images embedded into paper that are generally unnoticeable but become visible when shedding light through them. They serve primarily to identify manufacturers of paper. Historians find use in watermarks to recover information about the examined document's context, like its approximate manufacturing period and place of production. When encountering a new document with a watermark, it is thus crucial to match it to a similar watermark whose meta-information is already known to identify the artefact [1]. Hence, museums around the world store sizeable archives of them that have to be searched manually. This process is time-consuming, hindering cooperation with other institutions, and, therefore, this necessitates a universal watermark similarity-matching system that would enable historians to work together across the globe and reduce the time needed for this task tremendously.

In the context of watermark similarity matching a key part of the task is binarisation, a technique aiming to identify which parts of the scan are watermark segments and which - background. However, most such techniques are imperfect and would produce outlines with gaps (holes in the watermark curves wrongly classified as background) and noise (parts of the image wrongly classified as belonging to the watermark). This is why there is a need for a curve reconstruction module that can correct these discrepancies.

This paper aims to analyse what techniques can serve to reconstruct the watermark curves' shapes after the binarisation

techniques are applied. It aims to answer the question:

*“How can curve-reconstruction and hole-filling algorithms and models for binarised images aid watermark harmonisation to improve similarity matching and reduce user input?”*

To this end, a system must find how missing curve segments from the binarised image can be detected and reconnected. It must be able to approximate all types of curves. Such a module must be able to differentiate between curves and noise. It must also attempt to reconnect and adjust curves based on the way humans visually perceive shapes, connectedness, and smoothness as watermarks generally involve smooth curved shapes due to the bent wires used to imprint them.

In this paper, several methods that address this problem are presented, analysed, and compared to assess whether any approach would be effective in the watermarks domain.

## 2 Background

A curve is the image of an interval to a topological space by a continuous function. Watermark curves are continuous and planar. They can be non-bent (curves for which there exists a given direction for which the approximating points' projections would remain in the same order), bent (curves for which the afore-mentioned direction does not exist), self-intersecting, open (has endpoints) and closed (has no endpoints).

The system developed should be able to reconstruct curves of all types, as watermark outlines have a wide variety and often combine multiple such curves. To represent them, hence, a valuable basis would be Bézier curves [2], B-splines [3; 4], and NURBS curves [5] for open curves of all types and Catmull-Rom, also known as Cubic Hermite, splines [6] for closed ones, as watermark curve segments are continuous and polynomially-defined.

Bézier curves are parametric curves whose parts are influenced by control points that attract parts of the curve closer to them with differing force and area of influence. The process of generating them can be visualized using the de Casteljau algorithm [7], which recursively breaks down the curve into linear interpolations between control points. At each step, points are linearly interpolated. This breaking down continues until only a single point remains, which traces out the Bézier curve as a parameter  $t$  varies from 0 to (defining the  $t$ -values of the curve, tracing it from end to end).

B-splines (Basis Splines) and NURBS (Non-Uniform Rational B-splines) are extensions of Bezier curves that offer greater flexibility and control. B-splines are constructed from several polynomial segments, each influenced by a subset of control points (thus also requiring a knot vector). They are useful to get local control so that augmenting a given control point would influence only a section of the curve, unlike Bézier curves where changes can affect the entire curve. NURBS expand on this by adding weights to control points to model more complex shapes.

Finally, Catmull-Rom splines are similar but for them, the curve passes through the control points and the tangents are calculated using adjacent points. Their benefit is that they do

not require knot and weight vectors so they are simpler for curve generation.

### 3 Related Work

In 2023, Bantă et al. propose a watermark similarity-matching system trained and tested with scanned historic watermarks from the German Museum of Books and Writings [8]. The proposal involves a pipeline including a harmonisation, feature extraction, and similarity matching module. Harmonisation includes denoising and binarisation. Denoising takes an image as an input and attempts to increase the contrast between the watermark and the background, blur out noise, and remove lines. Binarisation segments the watermark as a white foreground and the rest as a black background. Then, a feature extraction module condenses this representation into a vector, and finally, a similarity matching module matches those vectors to one another based on a similarity metric. A significant drawback of this approach is a so-called binarisation post-processing sub-module where users receive the system's output and correct it manually by reconstructing curves and deleting noise in the binarised image. Requiring a user to draw is, however, unreliable and time-consuming. Thus, automating that section of the pipeline and limiting user input would significantly ease the potential users of such a system.

To aid the binarisation for watermark similarity matching, there is a need to automatically reconnect curve segments in the binarised images. This curve reconstruction can be done with three distinct approaches.

The first, curve approximation, includes sampling points from the curve and creating a fit for them. Graph-based methods, a sub-category of these approaches, structure the points in graphs and then apply given criteria to filter the outline [9]. A prominent example of feature-size criteria methods, another sub-category, is the Crust algorithm [10]. It takes as input a set of points in an image and splits the image into regions so that each region includes only one of those points and for any given region none of the points in it lies closer to another centre of a region than the one it is already in. This is the so-called Voronoi diagram with Voronoi vertices [11]. Then, the Delaunay triangulation [12] is computed on the union of the points and Voronoi vertices, which subdivides their convex hull into triangles whose circumcircles do not contain any of these points. Then, for each edge of the Delaunay triangulation if its two endpoints are in the original set of points this edge is added to the final curve. This technique is relevant for the case of watermarks as it allows for the reconstruction of multiple curves with non-uniformly sampled points from the curves which would be suitable for the problem case where curve segments of unknown lengths are missing. Its sampling condition, however, requires a sampling angle of at least 150 degrees between adjacent edges so it is not suitable for sharp edges and requires saturated sampling of smooth regions with bends.

Improvements to this method exist that can reduce the angle condition to less than 90 degrees as small as 60 degrees (HNN-Crust or Half Nearest Neighbour Crust) [13]. HNN-Crust achieves this by sampling more sparsely where possible

and connecting non-endpoints to both the nearest neighbour and the half-nearest neighbour which is the closest sample in the half-space  $H$  which is partitioned by the perpendicular bisector of the edge to the nearest neighbour and does not contain the nearest neighbour point.

Other curve approximation methods focus on noisy sampling fitting, sharp corners, minimising the total curve length (travelling salesman problem), and non-manifoldness (self-intersections of the curve) [9]. Relatively new HVS-based approaches (attempting to reconstruct the Human Visual System) aim to emulate how human eyes perceive and group visual elements, basing themselves on subsets of the six Gestalt laws: similarity, continuation, closure, proximity, figure-ground, and symmetry & order [14]. Examples of them include the DISCUR [15]. It can reconstruct multiple open or closed simple curves with sharp bends. Nevertheless, this heavily relies on an appropriate sampling of interior curves. Additionally, it requires a high sampling density of sharp bends which is difficult to obtain in the watermarks problem. VICUR, an improved variant, is also based on proximity and continuation [16]. Nevertheless, a major drawback to it is that it is highly sensitive to the user-tuned parameters. Non-uniform sampling can, hence, not be a good initialisation for such approaches.

The second approach, curve welding, attempts to extract information from the curve shapes themselves and tries connecting closely oriented segments. Such approaches also may rely on Gestalt laws [14] for how humans perceive connectivity but mostly rely on proximity and orientation of curve endpoints, ignoring other curve information.

The final approach, machine learning, formulates the task as an Image-to-image translation problem. This would require the input of an image that would be in one domain space (in this case curves with gaps) and translated into an image of another domain space (curves without gaps). Models that generate output like that are called generative models. They treat the output pixels as conditionally independent from all others given the input image [17]. Using such a model would enable a method to abstract itself from individual inputs and user-defined parameters and sampling and will depend merely on the dataset collection. In generative adversarial networks (GANs), a generative model is faced against an adversary discriminator which learns to determine whether a sample is from the target distribution or not [18]. Prior research has conditioned GANs on all types of multimedia like text and images [19; 20; 21]. This could be done in a paired or unpaired manner. In the former, the input is presented as pairs of input and target images. This gives substantial and conditioned information to the system to draw conclusions for the classification sets. An example of this class of models is Pix2Pix [17]. The key advantage of these methods is that they can identify defining features from a class very accurately because they are presented pairs of data. Nevertheless, acquisition of such data may not always be possible. In the latter sub-category, unpaired approaches, the input is just a set of images and the model must learn to differentiate the characteristics of classes, like in CycleGAN [22]. It can handle large amounts of data and quickly give results. Nevertheless, a key drawback of such methods is that the classes one aims

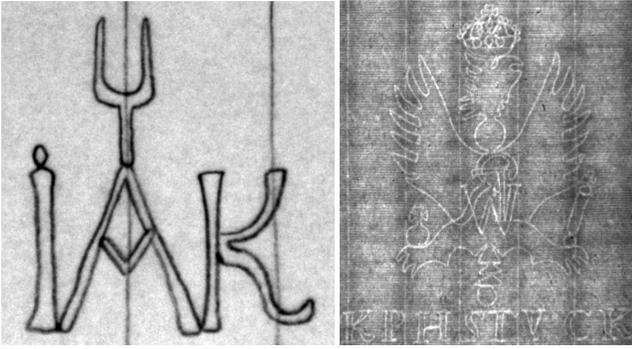


Figure 1: Example of a traced watermark, a watermark that has been manually traced with a pencil (left) and an example of an untraced watermark, a raw scan of the watermark on the paper as is naturally seen, usually with increased intensity contrast for better visibility (right)

to differentiate between need to have significant differences that can be detected through the analysis of the set fed to the system. This poses significant challenges in the learning of slight differences in similar classes, as could be the case for curves with and without gaps in the same conditions (binarised images).

None of these techniques have been examined in the context of curve reconstruction for binarised historic watermark scans. This paper will study the effectiveness of some of them and see how their advantages can be combined.

## 4 Methodology

To see how curve reconstruction can be applied to aid the historic watermark similarity-matching process, the problem was extracted from the domain of binarised historic watermarks to more general cases to see how robust the method would be on all curve types. The abstraction to general cases is detailed in Section 4.1. Then algorithms were selected and applied to those abstract cases for parameter optimisation as detailed in Section 4.2. Finally, a comparative quantitative evaluation metric was chosen to compare the approaches and draw conclusions about the final method. This is explained thoroughly in Section 4.3.

### 4.1 Dataset Creation

Watermarks are mostly imprinted using bent wires (hence, smooth and mostly continuous curves). Additionally, they are often traced with pencil by historians, making them similar to sketches (see Fig. 1). Hence, an assumption was made that abstracting from the problem to view general singular curve cases and sketches would be useful for creating a robust curve reconstruction algorithm.

Curves may exhibit many kinds of behaviour as explained above. To accommodate for curves of varying complexity that represent all of these characteristics two datasets of singular curves were, hence, created. The first included only open curves. 20 numbers between 0 and 100 were generated and then this was used as the number of randomly-generated control points for the curve. Then, open Bézier curves were produced from the control points. Each class of a generated

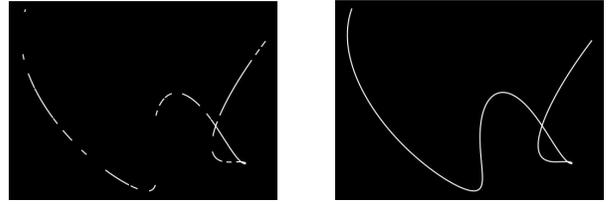


Figure 2: Example of an automatically-generated open curve (curve with endpoints) with automatically-generated gaps (left) and its original form before the gaps generation (right). The white pixels of the image denote the curve and the black ones - the background. This curve is a part of the dataset of open curves used for method tuning and validation.

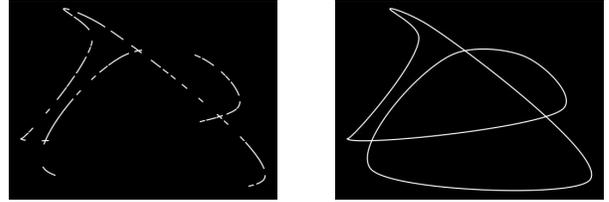


Figure 3: Example of an automatically-generated closed curve (curve without endpoints) with automatically-generated gaps (left) and its original form before the gaps generation (right). The white pixels of the image denote the curve and the black ones - the background. This curve is a part of the dataset of closed curves used for method tuning and validation.

number of control points got 50 generated samples, making the final number of open curves 1000 (see example in Fig. 2). This was chosen to ensure the representation of all curve behaviour permutations while also not making the training of a machine learning model too slow.

After this, 1000 closed curves were generated in the same way but using Catmull-Rom splines (see example in Fig. 3).

Then, since some watermarks are often traced, and drawn by people with a pencil, a dataset of human sketches was taken from TU Berlin and used as a more advanced and similar domain to the watermarks [23]. 20,000 images were taken from there and reformatted to be the same size as the open and closed curves (see example in Fig. 4). These 20,000 images were the entire set taken.

Finally, 311 binarised watermark images were taken from

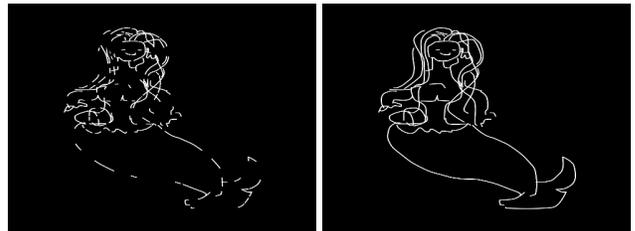


Figure 4: Example of a sketch with automatically-generated gaps (left) and its original form before the gaps generation (right). The white pixels of the image denote the sketch curves and the black ones - the background. This mermaid sketch is part of the dataset of sketches used for method training.

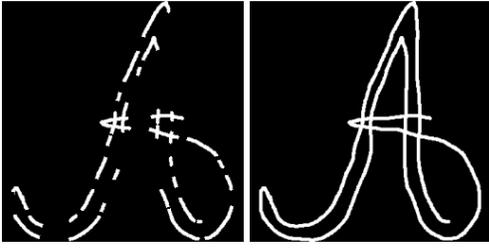


Figure 5: Example of a binarised watermark scan with automatically-generated gaps (left) and its original form before the gaps generation (right). The white pixels of the image denote the watermark and the black ones - the background (hence, binarised). This watermark is part of the dataset of sketches used for method training.

the German National Museum of Books and Writings. They included only traced watermarks, containing a single symbol, as only they could be reliably automatically binarised with the system proposed by Bantă et al. [8]. They were processed by the proposed pipeline, cropped around the white pixels, and fitted to be the same size as the other images (see example Fig. 5).

For each of those four categories gaps were generated. This, however, had to be done in a manner that would still enable the system to reliably reconstruct the general curve shape, as deleting keypoints (e.g. knots, endpoints, bends, local extrema) would significantly impede the reconstruction process. In those cases, even humans would hardly accurately be able to reconstruct those parts otherwise. This is done to examine how the method would perform under optimal conditions. To do this, each curve was skeletonized (made as thin as possible without affecting its continuity) and Hessian corner detection was applied to identify the keypoints.

Hessian corner detection is a method used to detect corners in an image using the Hessian matrix, a matrix of second-order partial derivatives of the image. Then, a radius was defined around each of the identified corners for curve pixels to be preserved. In this case, a radius of five pixels was used as images were of sizes  $496 \times 369$  and five pixels is enough to create a visible area around a given keypoint. All other curve pixels were deleted to represent the gaps. Examples of generated curves and their corresponding equivalents with gaps are visible in Figures 1, 2, 3, and 4 for all the classes. Finally, all of these pairs of images were split into 85% training and 15% testing sets using the stratified sampling technique, meaning that the split was done not on the whole set of all samples but on every sub-class (in this case, every sub-folder). All of these datasets are accessible in the project repository<sup>1</sup>.

## 4.2 Algorithms Choice

For the final method, a hybrid approach between machine-learning and analytical approaches was devised. To exemplify how it affects an input and how curve reconstruction fares with it, a butterfly sketch was chosen (see Fig. 6).

<sup>1</sup>[https://gitlab.ewi.tudelft.nl/cse3000/2023-2024-q4/Skrodzki\\_Castaneda/vpetkov-Automated-processing-of-scanned-historic-watermar.git](https://gitlab.ewi.tudelft.nl/cse3000/2023-2024-q4/Skrodzki_Castaneda/vpetkov-Automated-processing-of-scanned-historic-watermar.git)

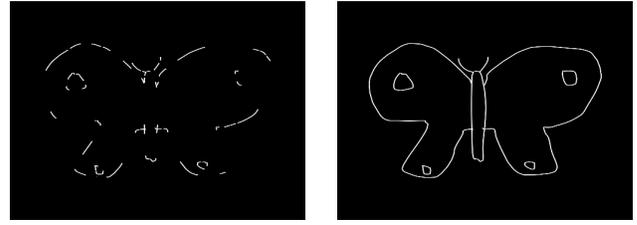


Figure 6: Example of a sketch that will be used to follow the effects of the hybrid approach on an image input. The input, image of the sketch with gaps, can be found on the left and its ground truth with no gaps can be found on the right.

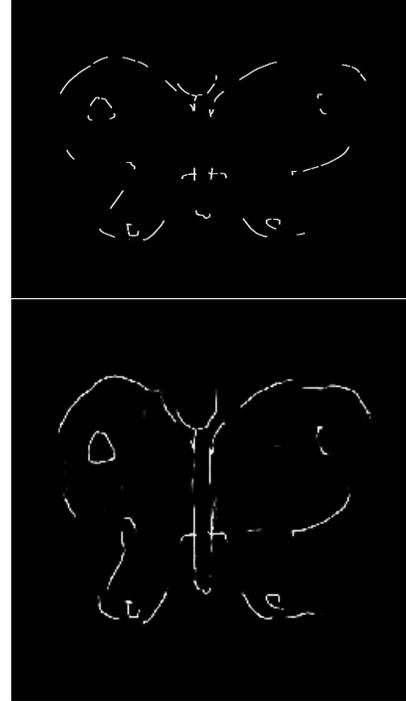


Figure 7: Example of an input, fed to the hybrid approach (up) and the output that the Pix2Pix step of this method produces (down).

Pix2Pix and cycleGAN were selected for the machine-learning methods as they represent the two classes of image-to-image translation problems: paired [17] and unpaired [22], respectively. Then, repeated reconstruction (in five iterations) using a model trained was performed to maximise results. Due to unnoticeable changes qualitatively, cycleGAN was not adopted for the final approach and only Pix2Pix was selected as the first step of the hybrid method (with five iterations performed). It resizes the image to  $256 \times 256$  for ease of processing. Its effect on the input can be seen in Fig. 7.

Then, the predicted output is fed into the analytical module. For it, experiments were done with Crust and VICUR. For VICUR, uniform point sampling with respect to the pixel distances was performed before the technique was applied. This, unfortunately, exhibited very poor results and coarse approximation on training images. Therefore, CRUST was prioritised for the final approach despite its inability to con-



Figure 8: Output that the Pix2Pix step of the hybrid approach produces for the example from Fig. 6 (left) and output after this result is processed by the Crust step of the method (right).

nect self-intersections. 250 pixels are sampled in total with each connected component contributing proportionally to its size. Within the segments, pixels are sampled uniformly to represent their parts of the image optimally. The algorithm’s impact can be seen in Fig. 8.

Finally, to do minimal shape optimisation, a simplistic welding method was devised. Reconstructable curves mostly require segment endpoints to have similar alignment and proximity. Additionally, two segment endpoints can be combined to be joined if they are not connected and lie in the image sectors defined by a certain angle offset on both directions of the point tangent in the opposite direction of the segment the points lie on (in this case 60 degrees on both sides). This is done by *lies\_within\_sector* in Algorithm 1. The final parameters used for the simplistic approach were 40 pixels distance, 60 degrees orientation threshold and 60 degrees sector threshold.

In this algorithm *compute\_image\_gradients(image)* computes the Sobel gradient (or first derivatives in the x and y direction) of the image. The method *compute\_orientations(grad\_x, grad\_y)* computes the angle of orientation of each pixel on the image by taking the 2-quadrant arctan  $\left(\frac{grad_y}{grad_x}\right)$  which returns the angle orientation within the range  $[-\frac{\pi}{2}, \frac{\pi}{2}]$ . Then *extract\_endpoints* takes the image, skeletonises its curves, and identifies points with specific geometric configurations of neighbours that correspond to endpoints and then returns their coordinates. Then, *combine\_connected\_endpoints(endpoints)* checks which endpoints are connected (checks if there exists a path of neighbouring white pixels) to one another on the image and groups them, returning a list of lists of pixel coordinate pairs. *angle\_between* calculates the angle between the two points’ orientations and *euclidean\_distance* - the Euclidean distance.

Finally, the Bezier curve is constructed using these parameters by taking the 2 endpoints and making 2 control points using the formulas  $P_1 = P_0 + \alpha \cdot T_0$  and  $P_2 = P_3 - \beta \cdot T_1$  where  $P_0$  and  $P_3$  are the endpoints and  $T_0$  and  $T_1$  are the corresponding tangent vectors.  $\alpha$  and  $\beta$  denote the weights of the tangent vectors for  $P_1$  and  $P_2$ , respectively (they control the distance of the control points from the endpoints). *num\_points* denotes the number of points the curve will return, calculated by constructing a linear space from 0 to 1 and

dividing it into this many sub-parts before using those values as t\_values for the Bézier curve.

---

#### Algorithm 1 Curve Segment Welding Algorithm

---

**Require:** image, distance threshold, orientation threshold, sector angle  
 $grad_x, grad_y \leftarrow compute\_image\_gradients(image)$   
 $orientations \leftarrow compute\_orientations(grad_x, grad_y)$   
 $endpoints \leftarrow extract\_endpoints(binary\_image)$   
 $segments \leftarrow combine\_connected\_endpoints(endpoints)$   
 $pairs\_to\_join \leftarrow \emptyset$   
**for** each pair  $pair\_1$  in  $segments$  **do**  
  **for** each endpoint  $point_x$  in  $pair\_1$  **do**  
    **for** each non-visited pair  $pair\_2$  in  $segments$  **do**  
      **for** each endpoint  $point_y$  in  $pair\_2$  **do**  
         $angle \leftarrow angle\_between(orientation(point_x), orientation(point_y))$   
         $distance \leftarrow euclidean\_distance(point_x, point_y)$   
         $lies\_within\_sector \leftarrow is\_within\_sector(point_x, point_y, tangent, sectorangle)$   
        **if**  $point_y$  is not visited already and  
           $distance < distance\_threshold$  and  
           $angle < orientation\_threshold$  and  
           $lies\_within$   
          **then**  
            add the pair with minimal angle difference  $(point_x, point_y)$  to the  $pairs\_to\_join$   
          **end if**  
        **end if**  
      **end for**  
    **end for**  
  **end for**  
**end for**  
reconstruct each pair by building a cubic Bézier curve between the endpoints with no kinks for a smooth transition between their orientations ( $\alpha=5, \beta=1, num\_points=100$ ) if it would not intersect more than 2 pixels from the existing curves  
display reconstructed curve

---

The output of the hybrid approach is then found after this third and final step.

### 4.3 Evaluation Metric Choice

Five metrics were used to evaluate how effective the approach is in curve reconstruction.

The first metric, intersection over union (IoU), considers what area is precisely coloured as part of the union of the ground truth and output [24]. This metric was selected as it indicates exact overlapping. It does not, however, reflect pixels approximately correctly matched by the system but not exactly. Equation (1) below, where A and B are the two curve sets identified showcases how it can be calculated.

$$IoU = \frac{\text{Area of Intersection}}{\text{Area of Union}} = \frac{|A \cap B|}{|A \cup B|} \quad (1)$$

The second metric, Hausdorff distance, denotes what is the maximal distance in pixels between the reconstructed curve

and ground truth [25]. This is useful as it showcases how mismatched the curves may be. This metric can be calculated with the equation (2) below where  $d(a, b)$  denotes the Euclidean distance between two points and A and B denote the sets of points of the curves of the predicted output and the ground truth.

$$H(A, B) = \max \left\{ \max_{a \in A} \min_{b \in B} d(a, b), \max_{b \in B} \min_{a \in A} d(b, a) \right\} \quad (2)$$

The final three metrics reflect that when an algorithm is applied it produces a prediction image where some pixels are visibly correct to a human but not fully identical to those of the ground truth reconstructed curve. To reflect that such results shall be considered correct, I consider a pixel from the ground truth to be captured correctly if there exists a pixel within a certain radius (five pixels in this case) from it in the predicted image that is also white. This is what we refer to as true positives (denoted as TP). Similarly, if such a pixel does not exist in this radius we have a false negative (denoted as FN). False positives (denoted as FP) are pixels in the predicted image that don't correspond to curve pixels in the ground truth. For this project, the precision, recall, and F1 score [26] were examined and calculated, as per equations (3), (4), and (5), respectively:

$$\text{Precision} = \frac{TP}{TP + FP} \quad (3)$$

$$\text{Recall} = \frac{TP}{TP + FN} \quad (4)$$

$$F1 = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \quad (5)$$

## 5 Experimental Setup, Results, and Discussion

The final hybrid method was trained on a set of 800 sketches and watermarks for time efficiency (to last only a few hours and not days). The set included all of the watermark training images and the rest were sketch images randomly drawn from the dataset. Then the model to use for the Pix2Pix was generated and used for the hybrid approach. For results reproducibility, this model was stored and is accessible in the project repository<sup>2</sup>.

The evaluation was run using the hybrid approach, compared to the Pix2Pix and Crust steps alone to assess the system and weigh its advantages and disadvantages. Each of those methods was evaluated on the training and testing set.

For the training set, for runtime reasons, 100 sketches from the sketches training set were randomly selected, as well as 50 watermarks from their respective training set. Then the method was run on all of them and scores were generated for the sketches class alone, watermarks class alone, as well as the total set of both classes. The scores for each of them included the intersection over union (IoU), Hausdorff distance, Precision, Recall, and F1 score as detailed in Section 4.3.

<sup>2</sup>[https://gitlab.ewi.tudelft.nl/cse3000/2023-2024-q4/Skrodzki\\_Castaneda/vpetkov-Automated-processing-of-scanned-historic-watermar.git](https://gitlab.ewi.tudelft.nl/cse3000/2023-2024-q4/Skrodzki_Castaneda/vpetkov-Automated-processing-of-scanned-historic-watermar.git)

Similarly, for the testing set, 102 sketches were taken and all 48 watermarks from the testing set to have 150 samples total in both cases. For reproducibility, each of the files taken for the evaluation, its generated pixel coordinates for the Crust step, as well as its evaluation scores have been recorded and are accessible in the project repository.

### 5.1 Results

The results of the hybrid, Pix2Pix, and Crust methods can be seen in Figures 9, 10, and 11, respectively.

When it comes to the first metric, intersection over union, all three methods show poor results for all sample categories with none of them surpassing a 50% overlap. This is, however, to be expected as the full overlap of the predicted output with the ground truth is nearly impossible even for a human. Thus, all techniques used rather aimed at approximation and not exact overlap.

Next, the Hausdorff distance metric reflects how far in pixels the most mismatched point of the curve is. Here, the watermarks class has significantly better scores than the sketches, meaning that the system is less likely to distort the curve with them, possibly due to their compactness and abundance of keypoints that led to those parts being more preserved in the gap generation process than with the sketches. The hybrid approach and Pix2Pix score similarly on this metric, whereas Crust has much higher distances, hinting at its inaccuracy when used alone.

Finally, when it comes to the last three metrics they are very much interrelated. On the one hand, recall shows whether the system introduces too much additional noise and false parts to the curve. Precision, on the other hand, examines how well the approximation of the ground truth worked. The F1 score in the end strikes a balance between them.

Recall scores across all three approaches remain similar with the minimal score recorded being 0.97. This means that less than 3% of the reconstructed curve was too far from the original curves, or in other words, represented noise. Therefore, all techniques used were successful in curve approximations without introducing erroneous artefacts.

Precision scores can be considered the most defining difference in the three approaches. In the Crust method, none of the categories surpasses 0.5, meaning that less than half of the ground truth curve is matched by the predicted output. This renders Crust not particularly useful for curve reconstruction of watermark scans. As for the hybrid approach, precision for the training samples is expectedly slightly higher than for the testing ones. The hybrid approach achieves as much as 71% precision for sketches from the training set and 66% for those in the testing set, meaning that for sketches it accurately approximates more than two-thirds of the original sketch. Nevertheless, results for watermarks are slightly worse with 58% and 54% for the training and testing set respectively. This renders the hybrid approach also not particularly useful in curve approximation in the watermarks domain.

Most importantly, however, results from Pix2Pix alone show that 83% of the watermark ground truth is correctly matched by the Pix2Pix method alone for the samples of the testing set and the system achieves as much as 91% precision for watermarks from the training set. This qualifies Pix2Pix

alone as a viable curve reconstruction approach for the watermarks. Regarding sketches, however, it shows similar but a bit better results from the hybrid approach, just below 75%.

The effects of the three approaches can also be observed on an example watermark of an outline and three letters in Fig. 12. There, Crust fails to approximate both the watermark outline and the letters. The hybrid approach manages to rectify this by having a fully reconstructed outline but missing sizeable letter segments. The pure Pix2Pix approach, on the other hand, manages to reconstruct everything almost fully, when compared to the ground truth. This example only demonstrates the potential of this method in this domain.

Considering the above, the F1 score only confirms what the precision score already pointed out: the proposed hybrid approach fails to reliably reconstruct the watermark curves. This can be explained by several shortcomings.

## 5.2 Discussion

Firstly, the assumption was made that analytical curve reconstruction approaches can be helpful to machine learning methods. Nevertheless, during the choice of a suitable analytical approach some of Crust’s key downsides were disregarded in the hopes of accurate results. Those included the step’s inability to connect non-manifold edges and deal with self-intersections, as well as the minimum required angle. This likely disconnected areas correctly connected by Pix2Pix already and/or wrongly connected them.

Secondly, a key shortcoming in the whole system is that the outcomes of the methods are all context-dependent. Therefore, any promising results of the methods above can be viewed as such only in the context of reconstructing curves in the given domain. The key assumptions that were made here included that there is barely any noise around the original curve which is highly unlikely in a realistic scenario of a watermark similarity matching system. Additionally, it was assumed that the keypoints of a given curve (endpoints, local extrema, self-intersection knots, and inflection points) would remain in the input unchanged and gaps would be made only between them. This assumption is also highly unrealistic but the lack of a viable existing system for watermark similarity matching left no alternative to that.

Thirdly, the set of binarised watermarks used for training and evaluation was too small to make statistically reliable conclusions based on them. For images of such complexity and size of  $256 \times 256$ , thousands of training images are recommended. This could not be done, however, due to the need for manual annotation to cover the wide variety of watermarks that exist, specifically untraced ones. To compensate for this, sketches and automatically-generated curves were used. Additionally, it is recommended to train only on watermarks to achieve optimal results for their class since the ultimate aim of the system is to handle only watermarks.

Finally, the quantitative analysis is context-dependent and ignores the human factor of recognising shapes and similarities. For example, a section of a curve might be visually approximately correct to a human, but if it is six pixels distanced from its original correspondent segment then the precision metric defined for a radius of five pixels would incorrectly detect it as mismatched.

		IoU	Hausdorff	Precision	Recall	F1
Testing	Sketches	0.13	39.22	<b>0.66</b>	0.97	0.77
	Watermarks	0.12	<b>22.89</b>	0.54	1.00	0.68
	Total	0.13	34.00	0.62	0.98	0.74
Training	Sketches	0.14	33.15	<b>0.71</b>	0.98	0.81
	Watermarks	0.12	<b>20.86</b>	0.58	1.00	0.71
	Total	0.14	29.06	0.66	0.99	0.78

Figure 9: Evaluation results of the hybrid method on the training and testing set of sketches and watermarks. In bold are the highest scores of this method. Metrics used are: intersection over union (IoU) which ranges from 0 to 1 where 0 denotes no overlap of the curves and 1 - full overlap; Hausdorff distance (Hausdorff) which is in pixels and ranges from 0 to the maximal possible size within the images, in this case, 256, where the higher the distance, the higher the mismatch; precision, recall, and F1 score, which range from 0 to 1 where 1 is the best possible result, and 0 - the worst.

		IoU	Hausdorff	Precision	Recall	F1
Testing	Sketches	0.21	29.30	<b>0.75</b>	0.99	0.84
	Watermarks	0.19	<b>16.00</b>	<b>0.83</b>	1.00	0.89
	Total	0.20	25.04	0.78	0.99	0.85
Training	Sketches	0.21	31.88	<b>0.74</b>	0.98	0.83
	Watermarks	0.21	<b>13.90</b>	<b>0.91</b>	1.00	0.95
	Total	0.21	18.85	0.87	0.99	0.92

Figure 10: Evaluation results of the pure Pix2Pix method on the training and testing set of sketches and watermarks. In bold are the highest scores of this method. Metrics used are: intersection over union (IoU) which ranges from 0 to 1 where 0 denotes no overlap of the curves and 1 - full overlap; Hausdorff distance (Hausdorff) which is in pixels and ranges from 0 to the maximal possible size within the images, in this case, 256, where the higher the distance, the higher the mismatch; precision, recall, and F1 score, which range from 0 to 1 where 1 is the best possible result, and 0 - the worst.

		IoU	Hausdorff	Precision	Recall	F1
Testing	Sketches	0.09	46.34	0.49	0.97	0.63
	Watermarks	0.07	38.64	0.22	0.98	0.35
	Total	0.09	43.89	0.40	0.97	0.54
Training	Sketches	0.09	51.37	0.46	0.97	0.61
	Watermarks	0.07	34.15	0.24	0.98	0.37
	Total	0.08	38.89	0.30	0.98	0.44

Figure 11: Evaluation results of the pure Crust method on the training and testing set of sketches and watermarks. Metrics used are: intersection over union (IoU) which ranges from 0 to 1 where 0 denotes no overlap of the curves and 1 - full overlap; Hausdorff distance (Hausdorff) which is in pixels and ranges from 0 to the maximal possible size within the images, in this case 256, where the higher the distance, the higher the mismatch; precision, recall, and F1 score, which range from 0 to 1 where 1 is the best possible result, and 0 - the worst.

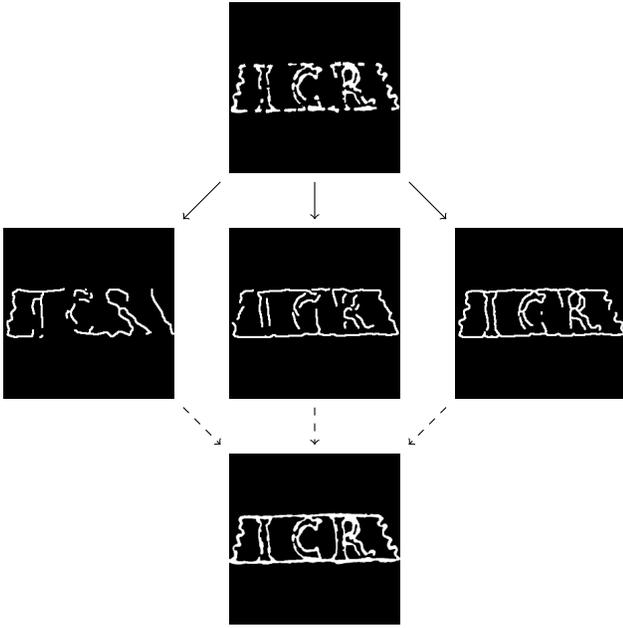


Figure 12: Visual comparison of the three approaches for an example binarised watermark scan. The input watermark with gaps can be found up, the outputs of the Crust, hybrid, and Pix2Pix approaches on this image input can be found in the middle row, next to one another, from the left to the right, respectively, and the ground truth without gaps those results could be compared to is below them.

## 6 Responsible Research

ChatGPT<sup>3</sup> was used to aid certain aspects of the programming. It was used to get ideas on which techniques may be effective for the problem, or what certain techniques meant, as well as what papers might be useful to read considering the problem. ChatGPT was also sometimes used for code generation for simple algorithms. It is important to note that all queries used for programming were to get an idea of how to approach a problem. Any code that was provided by ChatGPT was heavily altered to fit the context of this project. No code that was generated by ChatGPT was used in an unaltered form.

An open-source GitHub repository was used to gain insight into Pix2Pix, CycleGAN and their fundamentals<sup>4</sup>. Some code from this repository was used but a lot of it had to be adjusted to fit the curve reconstruction problem.

Similarly, another open-source GitHub repository<sup>5</sup> was experimented with to gain insight into multiple analytical curve reconstruction methods. Most notably, an executable file implementing VICUR and Crust was incorporated into the final method proposed.

Finally, watermark images have been provided in digitized form by the German Museum of Books and Writing for this project. Due to their age, the documents can also no longer be

<sup>3</sup><https://openai.com/blog/chatgpt>

<sup>4</sup>[https://github.com/bnsreenu/python\\_for\\_microscopists](https://github.com/bnsreenu/python_for_microscopists)

<sup>5</sup>[https://gitlab.com/stefango74/curve-benchmark/-/tree/master?ref\\_type=heads](https://gitlab.com/stefango74/curve-benchmark/-/tree/master?ref_type=heads)

copyrighted. It has been, thus, assumed that the museum has acquired these documents in an ethically responsible manner since it is an esteemed organization.

The same holds for the open-source sketches dataset<sup>6</sup>, collected by TU Berlin. The assumption is made that they do not contain any sensitive or personal information and that the creators of the sketches consented to their work being published online.

## 7 Conclusions and Future Work

In this paper, a new method for curve reconstruction in scanned historic watermarks was presented. It combined a conditional generative adversarial network (Pix2Pix) with an analytical feature-size criteria algorithm (Crust) and a Gestalt-law-based simplistic endpoints welding method. The system was developed and trained on the basis of a simulated environment of binarised grayscale images of automatically generated singular open and closed curves of varying complexity, human-drawn sketches of varying real-world shapes, as well as binarised watermarks scans to emulate the variety of shapes and curves in watermarks. The system was then trained on sketches and watermarks and tested on the same categories. The quantitative results on intersection over union, Hausdorff distance, precision, recall, and F1 score are then compared with the results of the machine learning and analytical components separately. The analysis shows promising results for the performance of conditional generative adversarial networks in the area of watermarks curve reconstruction but the proposed hybrid approach does not improve those scores.

This exemplifies the need for more research into the analytical component, as Crust is very simple and performs poorly. A possible substitute for it could be HNN-Crust [13], its improved version with a lower minimum angle. Methods that could be further integrated include GathanG [27] which can handle self-intersections, sharp bends, open curves, multiple components, and non-uniform curve sampling but not noise and outliers. An alternative that can deal with these is Peel [28], although it can not handle sharp bends and has a higher runtime complexity of  $O(n^2)$ . An interesting potential application can also be genetic algorithms like the Firefly algorithm [29] and generalised iterative methods like LSPIA [30] and NewtonGA [31]. Such methods start with an initial rough curve approximation and then iteratively move its control points and adjust its parameters until certain convergence criteria are met. Such approaches are, nevertheless, very sensitive to their initialisation and largely focus on singular curves which is undesirable in the given domain.

Additionally, more conditional generative adversarial networks can be leveraged against Pix2Pix to identify the best option from them for the watermarks domain.

Above all, more accurate dataset representation can be performed to abstract the method tuning and results from the initial assumptions made about the lack of noise in input curves, preservation and identification of curve keypoints, and sketches interchangeability with watermarks. That can be

<sup>6</sup><https://cybertron.cg.tu-berlin.de/eitz/projects/classifysketch/>

done by manually annotating a set of a few thousand watermark scans and running them through a finalised watermark similarity matching system to obtain their binarised curve equivalents with noise where keypoints might be sometimes fully omitted.

Finally, a prolific qualitative analysis of the effectiveness of curve reconstruction methods can be done through a user study to recognise the human perspective on the problem. This is vital as the method is to be introduced in recognition and similarity matching modules that aid humans, a task otherwise performed manually and visually.

## References

- [1] L. Müller, “Understanding paper: Structures, watermarks, and a conservator’s passion,” *Harvard Art Museums*, May 2021.
- [2] P. E. Bézier, “Mathematical and practical possibilities of unisurf,” *Computer Aided Geometric Design*, pp. 127–152, 1974.
- [3] C. de Boor, *A Practical Guide to Splines*. Springer-Verlag, New York, 1978.
- [4] R. Riesenfeld, “Applications of b-spline approximation to geometric problems of computer-aided design,” Ph.D. dissertation, Syracuse University, 1973.
- [5] K. Versprille, “Computer-aided design applications of the rational b-spline approximation form,” Ph.D. dissertation, Syracuse University, 1975.
- [6] E. Catmull and R. Rom, “A class of local interpolating splines,” in *Computer Aided Geometric Design*, R. E. BARNHILL and R. F. RIESENFELD, Eds. Academic Press, 1974, pp. 317–326. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/B9780120790500500205>
- [7] G. Farin, “Bezier and b-spline techniques,” in *Geometric Modeling: Algorithms and New Trends*, J. Hoschek and D. Lasser, Eds. Springer, 1993, pp. 35–46, includes discussion of de Casteljau’s algorithm.
- [8] D. Bantă, A. Lantink, V. Petkov, A. Marin, and S. Kho, “A watermark recognition system: An approach to matching similar watermarks,” 2023.
- [9] D. B. Blumenthal, T. Friedrich, and T. T. Saunderson, “Cage: Cluster-based approximation of graph edit distance,” *arXiv preprint arXiv:2103.095834*, 2021. [Online]. Available: <https://arxiv.org/abs/2103.095834>
- [10] N. Amenta, M. Bern, and M. Kamvysselis, “A new voronoi-based surface reconstruction algorithm,” in *Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques*. ACM, 1998, pp. 415–421.
- [11] G. Voronoi, “Nouvelles applications des paramètres continus à la théorie des formes quadratiques. deuxième mémoire. recherches sur les paralléloédres primitifs,” *Journal für die reine und angewandte Mathematik*, vol. 134, pp. 198–287, 1908, original paper on Voronoi diagrams.
- [12] B. Delaunay, “Sur la sphère vide,” *Izvestia Akademii Nauk SSSR, Otdelenie Matematicheskikh i Estestvennykh Nauk*, vol. 7, pp. 793–800, 1934, original paper on Delaunay triangulation.
- [13] S. Ohrhallinger, S. Mitchell, and M. Wimmer, “Curve reconstruction with many fewer samples,” *Computer Graphics Forum*, vol. 35, pp. 167–176, 08 2016.
- [14] K. Koffka, *Gestalt Psychology: An Introduction to New Concepts in Modern Psychology*. Harcourt, Brace and Company, 1935.
- [15] Y. Zeng, T. A. Nguyen, B. Yan, and S. Li, “A distance-based parameter free algorithm for curve reconstruction,” *Computer-Aided Design*, vol. 40, no. 2, pp. 210–222, 2008.
- [16] T. An Nguyen and Y. Zeng, “Vicur: A human-vision-based algorithm for curve reconstruction,” *Robotics and Computer-Integrated Manufacturing*, vol. 24, no. 6, pp. 824–834, 2008, fAIM 2007. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S073658450800046X>
- [17] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, “Image-to-image translation with conditional adversarial networks,” *CVPR*, 2017.
- [18] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, “Generative adversarial nets,” *NIPS*, 2014.
- [19] S. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, and H. Lee, “Generative adversarial text-to-image synthesis,” in *Proceedings of The 33rd International Conference on Machine Learning*, 2016.
- [20] H. Tang, D. Xu, N. Sebe, and Y. Yan, “Attention-guided generative adversarial networks for unsupervised image-to-image translation,” in *International Joint Conference on Neural Networks (IJCNN)*, 2019.
- [21] H. Tang, H. Liu, D. Xu, P. H. Torr, and N. Sebe, “Attentiongan: Unpaired image-to-image translation using attention-guided generative adversarial networks,” *IEEE Transactions on Neural Networks and Learning Systems (TNNLS)*, 2021.
- [22] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, “Unpaired image-to-image translation using cycle-consistent adversarial networks,” in *Computer Vision (ICCV), 2017 IEEE International Conference on*, 2017.
- [23] M. Eitz, J. Hays, and M. Alexa, “How do humans sketch objects?” *ACM Trans. Graph. (Proc. SIGGRAPH)*, vol. 31, no. 4, pp. 44:1–44:10, 2012.
- [24] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, “The pascal visual object classes challenge: A retrospective,” *International journal of computer vision*, vol. 111, no. 1, pp. 98–136, 2015.
- [25] M.-P. Dubuisson and A. K. Jain, “A modified hausdorff distance for object matching,” *IEEE Transactions on Pattern Analysis & Machine Intelligence*, no. 8, pp. 1546–1558, 1994.

- [26] C. J. van Rijsbergen, "Information retrieval," *Butterworths*, 1977.
- [27] T. Dey and R. Wenger, "Fast reconstruction of curves with sharp corners." *Int. J. Comput. Geometry Appl.*, vol. 12, pp. 353–400, 10 2002.
- [28] A. D. Parakkat, S. Methirumangalath, and R. Muthuganapathy, "Peeling the longest: A simple generalized curve reconstruction algorithm," *Computers amp; Graphics*, vol. 74, p. 191–201, August 2018.
- [29] A. Gálvez and A. Iglesias, "Firefly algorithm for bezier curve approximation," in *2013 13th International Conference on Computational Science and Its Applications*, 2013, pp. 81–88.
- [30] C. Deng and H. Lin, "Progressive and iterative approximation for least squares b-spline curve and surface fitting," *Computer-Aided Design*, vol. 47, pp. 32–44, 02 2014.
- [31] Q. Song and P. Bo, "Newton geometric iterative method for b-spline curve and surface approximation," *Computer-Aided Design*, vol. 172, p. 103716, 2024. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0010448524000435>