Research Article

Eugenia Sinatti*, Simon Weckert, and Ewelina Dobrzalski

Image Garden

Curating Collections and Designing Smart Exhibitions with AI-Based Tools

Abstract: Image Garden is an AI-based toolbox for curating collections and designing smart exhibitions that was developed by **ART+COM Studios** [ART+COM Studios. 2021. ART+COM. https://artcom.de/en/] in the research project **QURATOR** [QURATOR Bündnis. 2021. Qurator. https://qurator.ai]. This paper describes the application scenarios underlying the development of Image Garden. It also presents the technologies integrated in its processing pipeline, and sketches Show Cases that were implemented to demonstrate and evaluate functionality and usability.

Keywords: Digital Curation, Museum Collections, Smart Exhibitions, Machine Learning, Image Clustering, Dimension Reduction, Edge Detection, Colour Recognition, Latent Space

1 Introduction

Most museums are constantly trying to make larger parts of their collections easier accessible to a wider audience. Especially in the context of the pandemic it became an important aspect to enable the access to collections of museums while not being physically accessible to the public through the current restrictions. This involves, on the one hand, digitizing collection objects, including those not on display at the museum, and making them available on the Internet.

On the other hand, digital technologies are used to tell personalized stories about the objects – relating them to other objects in the collection, or to persons, places and events in the real world. This can be realized through i. e. the use of diverse screens or projections, VR or AR goggles, expanding exhibitions through tactile movement, using the devices of the visitors in the spirit of "bring your own device" or setting markers in the exhibition itself. In doing so, some museums are developing hybrid formats combining physical presence with remote access.

Image Garden aims at providing knowledge workers with an AI-based toolbox, which supports them in the complex tasks of curating collections or designing smart exhibitions.

2 Background and Application Scenarios

ART+COM, founded in 1988, designs and develops new media installations and spaces. We use new technology as an artistic medium of expression and as a medium for the interactive communication of complex information. In the process, we are improving the technologies constantly and exploring their potential for spatial communication and art. ART+COM creates exhibits for exhibitions, museums and brand spaces, designing and implementing media installations and spaces that impart complex content in a targeted manner and turn information into an engaging experience.

Exhibition design usually involves the analysis of a wide range of content with the purpose of selecting and combining content for specific exhibits. To support knowledge workers like curators, scientific teams, designers or communication experts in this process, ART+COM developed AI-based tools in the research project **DKT** (Digital Curation Technologies) [7]. A system based on Named Entity Recognition was used to extract information from texts, which was then mapped to an interactive Wikidata visualization displaying entities and relations between them as a knowledge graph [14]. This enabled a visual way of analyzing data and helps into new findings in different research fields.

Image Garden, developed in the context of the research project QURATOR, further extends these tools and focuses on image clustering for curating collections and designing smart exhibitions. The aim of the project lies in supporting knowledge-workers and editors in the complex work of curating digital content. This is realised through

*Corresponding author: Eugenia Sinatti, ART+COM Studios, Berlin, Germany, e-mail: eugenia.sinatti@artcom.de
Simon Weckert, Ewelina Dobrzalski, ART+COM Studios, Berlin, Germany, e-mails: sw@simonweckert.com, ewelina.dobrzalski@artcom.de

the automatization of curation-activities as well as the generation of digital contents via Machine Learning (ML) and especially Artificial Intelligence (AI). As the amount of digital content and networked information is growing, it seems urgent to develop better tools that can help to navigate through those large amounts of data. Image Garden explores the visual path of this challenge.

QURATOR is a joint research project, started in November 2018 and is funded by the Federal Ministry of Education and Research (**BMBF** [11]). The ten-member collective of QURATOR is made up of Berlin-based research centres *DFKI* and *Fraunhofer FOKUS*, companies *3pc*, *Ada Health*, ART+COM Studios, *Condat*, *Semtation*, and *Ubermetrics*, as well as *Wikimedia Deutschland* and the Prussian Cultural Heritage Foundation (*Staatsbibliothek zu Berlin*).

The tool Image Garden will also be evaluated in the context of **NuForm** (*Neue Formen der Begegnungskommunikation*) [29], a joint research project of ART+COM and the *Museum für Naturkunde Berlin*, that started in January 2021. NuForm focuses on hybrid formats combining physical attendance with remote access. The museum possesses a wide collection of items covering zoology, palaeontology, geology and mineralogy of highest scientific and historical importance. The access to the collection is mostly possible through physical presence. The project *NuForm* wants to find new ways of remote access with digital means.

Different approaches to enable a remote access to collections exist as more and more museums are digitizing their entire collections. Some are already offering access to the digitized content via their website, providing search engine-like interfaces: users can enter one or several keywords and as a result a list of collection objects is displayed, sometimes even with images and metadata.

However, there is often no connection between the real objects displayed on-site and the digital content on those websites. Moreover, the information is very object-centric and objects are presented separately. Neither are they shown in the context or in relation of a collection nor are they linked to persons, locations, events, etc. in the real world, which is giving an isolated perspective on the objects.

In contrast, Image Garden focuses on displaying objects in the context of the collection by clustering them according to visual similarity or metadata, as timelapses or other numerical aspects of the objects containing geographical aspects and others. This provides a basis for digitally curating the collection, as well as designing smart exhibitions, in which visitors are offered personalized tours and information. For the latter, relations between collection items and real-world knowledge, e. g. from Wikidata, can be used. Experiencing an exhibition that matches the interests of the given visitor enables a more immersive experience. The content is better understood and has a consistent impact on the visitor itself.

3 Image Garden Technology

Alexander von Humboldt published two dozen books and more than 700 articles and essays. It is less known that the naturalist and travel writer was also a draftsman and graphic artist: around 1,500 illustrations accompany his writings. His work has been collected in the book *"Das graphische Gesamtwerk"* of *Alexander von Humboldt* [33] mainly from Humboldt's world-famous American travel.

This work was a brilliant starting point for our project, but it turned out that no digitalized version of the images were available. We therefore had to get similar data of botanical illustrations from other sources to realize each step in the tool Image Garden. We found **the Biodiversity Heritage Library** on *flickr* [26], which has a vast collection of illustrations of over two Million objects. These images are identified and uploaded in bulk using an algorithm that offered us a great opportunity for serendipitous discovery of image datasets via browsing.

The data was first downloaded from the website, then the images were normalized to their resolution and each image in the metadata was checked for copyrights. Finally, we extracted 2,600 botanical illustrations with legal rights to use in our dataset.

Inspired by the botanic illustrations from *Alexander von Humboldt*, we wanted to build a search engine that was able to identify images by the gesture of a drawn line. The search was supposed to be rather playful so that the user could grasp the archive of botanical illustrations in a narrative way. The goal was to look for a single line in the dataset and show the image that contained a similar line to the searched line. This approach promised to be the simplest in its gesture and at the same time the most precise, as a single pixel could be used and translated through the connection to other pixel into a path of two dimensions.

The chosen approach of drawing a line also resembles the original process of the drawing itself and makes a poetic connection between the original work and the line search. Time and space become connected through the search and the approach reminds the user that the base of each drawing lies in a line.

To achieve our goal of structuring the data, we had to divide an image dataset into useful data and those not necessary – inspired by the image of a gardener who separates

weed from flowers – and find a method to curate an un-clean dataset in a clustered way.

3.1 Preprocessing

As starting point for Image Garden we made use of computation and interactive visualization by the embedding of the image collection in a 2 or 3-dimensional space via the tool **PixPlot** [35]. The main idea of the tool *PixPlot*, developed by **Yale Digital Humanities Lab Team** [36], is to vectorize each image of the dataset with the Convolutional Neural Network Inception V3, which is an application of **Keras**.[1] Running this application with the pre-trained weights for *ImageNet* [27], a dataset of over 14 Millions images, we were able to assign to each image of the collection an inception 2,048-dimensional vector, where each entry represents the discrete probability distribution over one of the 2,048 target classes for each class. I. e. the value in each entry is the probability that the image belongs to the particular target class of 2,048. The discrete probability distribution over the target classes is calculated via the *Softmax* activation function on the last layer of the convolutional neural network, which is existent usually in multiclass classification problems.

After that we applied the algorithm **UMAP** [25] (Uniform Manifold Approximation and Projection for Dimension Reduction) [17], which is on one side an algorithm to reduce the dimension of each vector of data points vector from the original set to handle with the data in a faster and more usable way. On the other side it gives each data point a position in the 2- or 3-dimensional space via complex mathematical computations, taking into account the correlation between the data-points. This means, that the local connectivity of the data points can be measured by correlation metrics, that describe whether or not a relationship between two data points exists.

The result of that process is the embedding of the image collection in a 2- or 3-dimensional space with an underlying structure of clearly recognizable clusters, which reflect the visual similarity between the images (See Figure 1). In this part of the process we made use of a very powerful and fast algorithm for unsupervised learning problems, in which the data is not labeled at the beginning of the process. After applying *UMAP*, we were able to subdivide the data in several well definable clusters.

Figure 1 shows some interesting examples of the results of clustering on the dataset of 50,000 images from the
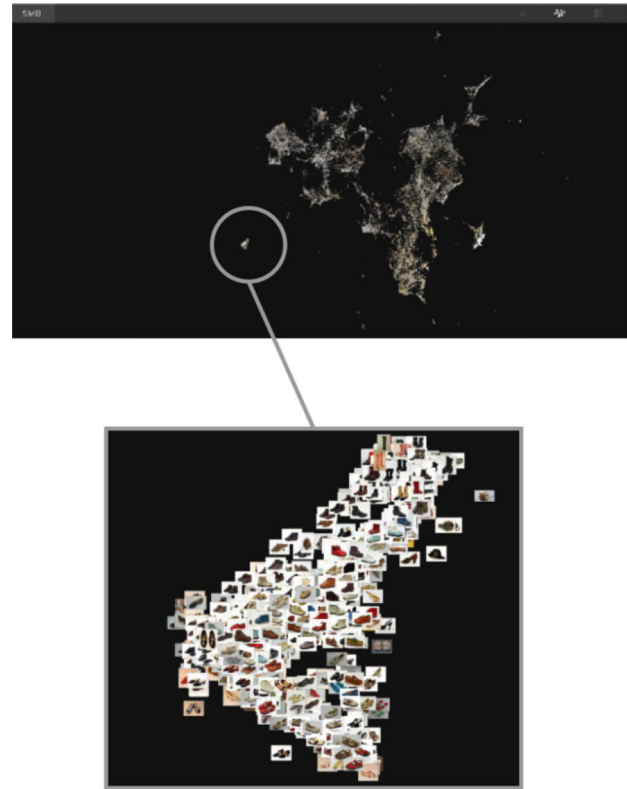

Figure 1: Cluster of several images of different styles of shoes. Created by ART+COM Studios.

collection of the *Staatliche Museen zu Berlin* (**SMB**) [18]. In the Figure 1 the clustering of similar objects shown on the images is clearly visible.

At the same time the visualization of the *UMAP* gave us the possibility to filter images during the preprocessing step, e. g. by quickly finding outliers or images not really belonging to the main subject of the chosen collection.

Figure 2 provides an overview of the pipeline used in Image Garden to process large collections of objects with images and metadata. First, the image data (and the metadata) had to be pre-processed, e. g. by normalizing the formats and adjusting their size to increase the performance, by removing artifacts, free-form selection etc. Then the parameters for clustering and for the use of the algorithm *UMAP* had to be configured.

After applying *UMAP*, an interactive visualization of all objects based on Java Script was generated with the help of *WebGL* for rendering interactive 2-dimensional and 3-dimensional graphics. That allowed us to use GPU-accelerated usage of physics and image processing and effects as part of a web page canvas. To compute the preprocessing and the visualization of the data, it took us one hour for 15,000 images.

---

1 Keras [15] is a deep learning API written in Python, running on top of the machine learning platform **TensorFlow** [31].
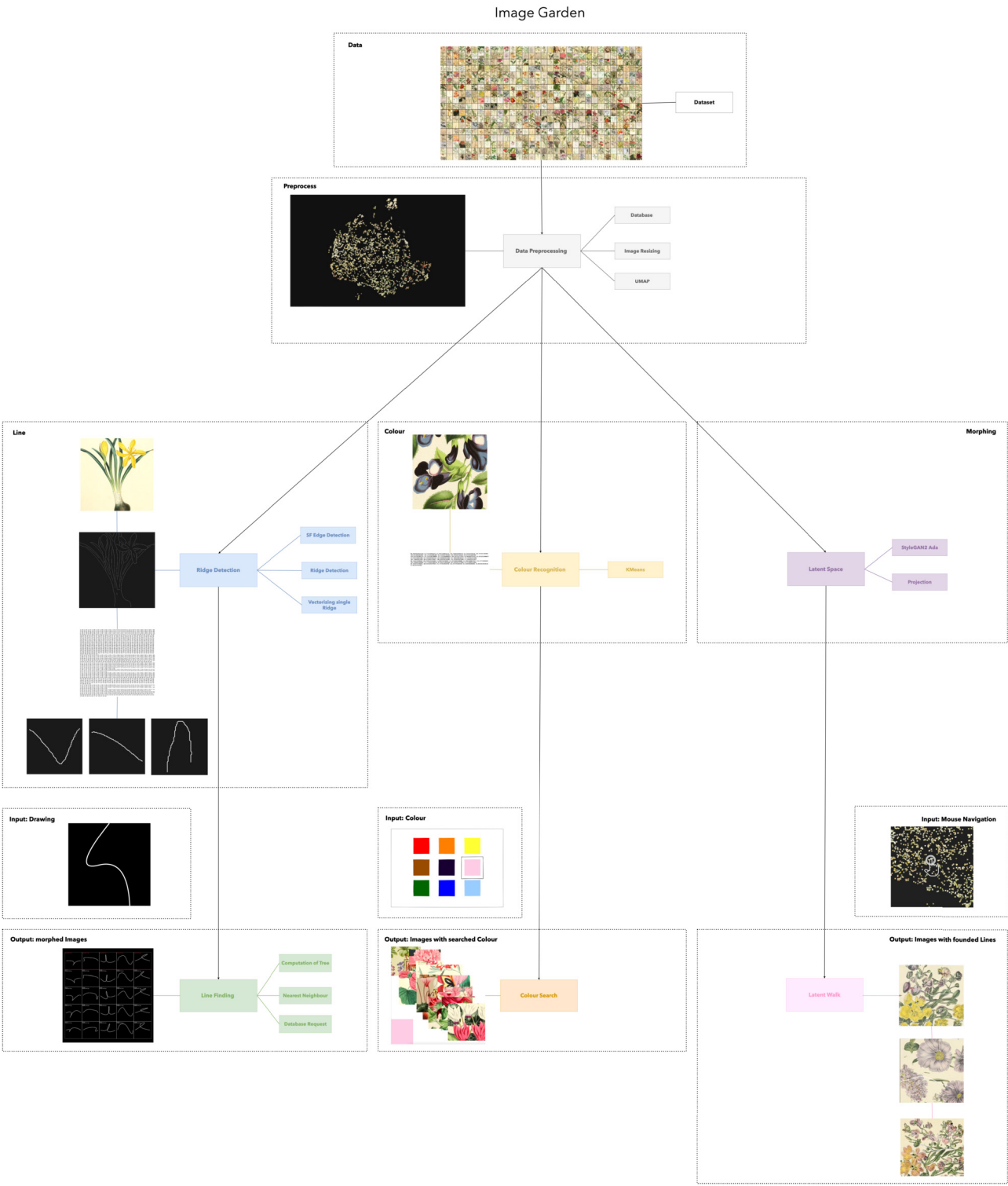
Figure 2: Pipeline of Image Garden. Created by ART+COM Studios.

3.2 Ridge Detection

Motivated by developing an image search by drawing lines, we next compared different algorithms for edge detection and decided to apply *Structured Forests* [8], since the found edges were cleaned from "salt and pepper"-noise and the time for the computation took only 30 seconds per image. To obtain even more clearly defined lines we applied **Ridge Detection** [10] on the edges of each image afterwards (See Figure 3). The computation of the ridges of a single image took around 50 seconds, which is a factor, that would need to be accelerated.

At the end of this process we had clear ridges and could continue in the process. The next step was to extract from every ridge detected image all lines to be able to apply later the approximate nearest neighbour (*ann-*) algorithm **Annoy** [9]. This allowed us to compare all lines to the drawn one.

For this process we wrote an algorithm, that translated each line of the image into a path of 2-dimensional coordinates, in which each coordinate denotes a white pixel of a line. To realize this, first we needed to iterate over each pixel of the image, analyzing each pixel neighbouring a white pixel. The concept is similar to the popular game of "Battleship": once you find a white pixel, you follow the white neighbour pixel, until you land in a black one. As a result you obtain a line of the image in the same way as a you find a ship in the game "Battleship". Finally each line will be translated as a path of 2-dimensional coordinates for drawing later an image of each line of the size of 200×200 pixel.

3.3 Nearest Neighbour

At this point of the pipeline we had to convert the wide dataset of 200×200 pixel images into feature extraction vectors, by slightly modifying the TensorFlow's original *ImageNet* script [34]. We used the convolutional neural network Inception, pretrained on *ImageNet* for classification and detection as mentioned in "Going Deeper with Convolutions" [30]. The script from *ImageNet* uses an image as an input and then returns a set of probabilities corresponding to the class labels as output.

To identify similar images we had first to work with float vectors, then with class labels. The labels are useful for tasks like image search but lack precision to identify similar images. Float vectors on the other hand, capture more data of the original object than labels. After obtaining the float vector from the penultimate layer of the neural network we were able to get the weights as vector represen-

Figure 3: Edge Detection via Structured Forest Algorithm and Ridge Detection. Created by ART+COM Studios.

tations of each image instead of a string class label by the last layer in the neuronal network. The result is more differentiated: an image vector can represent for instance 20 % a cat, 60 % a dog and 20 % others. This allows to perform a traditional vector analysis by using images.

Once we had the feature extraction vector files of each line, we started to look for a Nearest Neighbour algorithm to find the fastest and easiest way to calculate the distance for points in space that are close to a given query for a data point. We compared different *Metric Tree* methods like **Vantage Point Trees** [21] and **K-Dimensional Tree** [6]
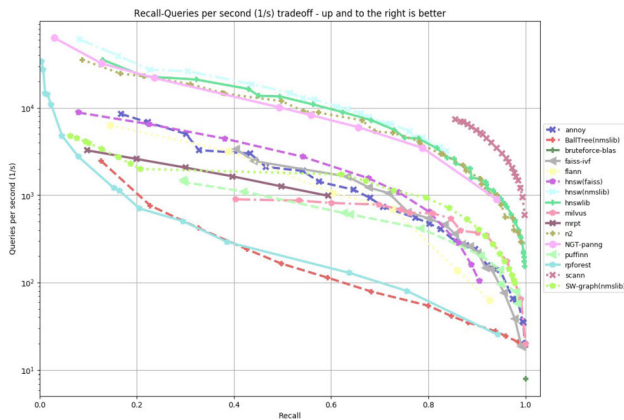
Figure 4: Ann-benchmark. Via **Spotify Annoy Github Repository**.

but decided to use an **Approximate Nearest Neighbour Search** [1] algorithm because of its multiple advantages.

As shown in the diagram in Figure 4, **ann-benchmark** [4] is a benchmark for several approximate nearest neighbour libraries and *Annoy* [9] seems to be fairly competitive, especially at higher precisions.

We picked Annoy because of its speed and its ability to use static files as indexes, that represent every data point. Through that we were able to share index across processes. Another advantage of *Annoy* is also its function to decouple created indexes from loading. This way we were able to use indexes as files and include them quickly into the database. As *Annoy* minimizes its memory footprint, the indexes are also quite small which in consequence was useful for multiple CPU computing, as we had to build the index just once.

With *Annoy* we build the Tree, where each node represents a line, to find the most similar line to the drawn one.

Figure 5 shows a result of the search in the collection by drawing a free line on the screen.
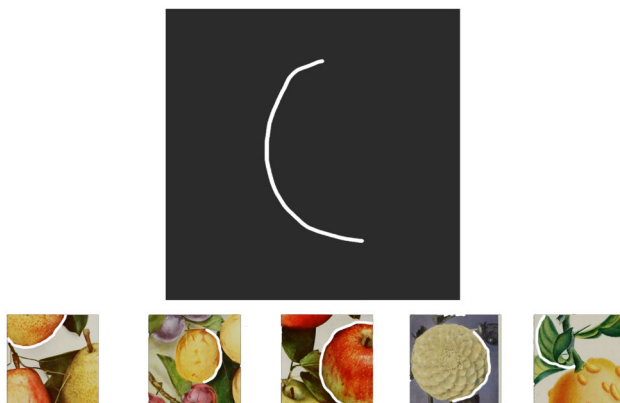
3.4 Colour Recognition

Image Garden offers also the possibility to navigate an image collection to search for a specific colour. To attain this, we used the open source unsupervised cluster algorithm **KMeans** [16]. First each image of the collection needed to be identified as a matrix with a smaller size of width and height. Each entry of the matrix is hereby a 3-dimensional vector, representing the colour of the pixel in the RGB (Red, Green, Blue).

We applied *KMeans*, which takes as a parameter the number of clusters, that are at the same time the amount of main colours to identify in the image. As output of this clustering we received clusters of similar colour in the image. The main colours of each image were the center of the computed clusters.

We wanted to create a search engine, that was able to find a colour, even if it was present in only a very small part of the image (See Figure 6). To obtain this we needed a format of each RGB vector, which represents each colour, to compare with the searched colour vector. We converted the RGB colour input and each RGB colour of all images into a colour vector of the **CIELAB** [3] colour space, in which each colour is a 3-dimensional space with floating numbers.

Finally we were able to search for a specific colour within the main colours of each image by calculating the Euclidean distance between two of them in CIELAB color space. We needed to give a threshold as parameter, which defines, how different the colours of each image and the searched colour can be. To find images with exactly the same colour or one very close to it, the threshold needs to be set really low.
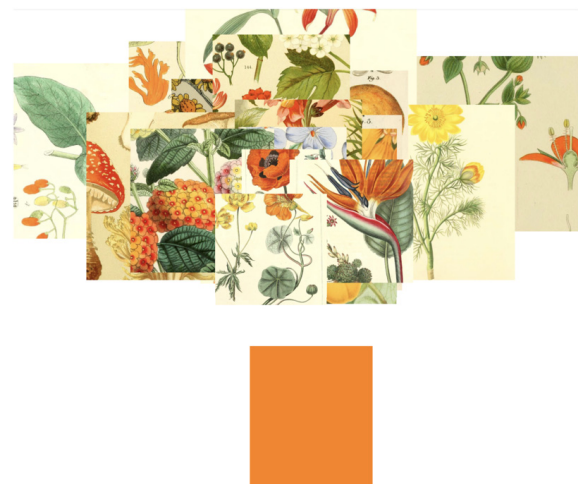

Figure 5: Example of a line drawing. Created by ART+COM Studios.

Figure 6: Example of a colour search. Created by ART+COM Studios.

3.5 Latent Space

In the last step of the pipeline the dataset of botanical drawings was used to generate morphed images of plants through a generative adversarial network [32]. This gave us a playful way to interact with the image collection and enabled us to generate new botanical drawings that have never been seen before.

As we had a very small dataset of only 2,600 drawings of botanical images we needed to find a way to expand the dataset to reach better training results for the neural network. The usual techniques used to expand such a small dataset are cropping, rotating, flipping, scaling, translating and adding Gaussian noise to the dataset. More advanced augmentation techniques use neural style transfer to mix the dataset in the style of another content (compare with [19]). The above step is a very time consuming technique, but luckily NVIDIA Research Projects released in the time of our research an updated version of **Style-GAN2** [24] called **StyleGAN2-ADA** (StyleGAN2 with adaptive discriminator augmentation) [23] which enabled us to use our current dataset without any data expanding techniques. Through the updated version the network generated significantly better results for datasets with less than 30,000 training images.

With this new version of *StyleGAN2-ADA* we were able to continue to the next step of the pipeline. For a playful interaction we used the visualization of the *UMAP* as an input interface and started to morph between the images. This was possible with the trained *StyleGAN2-ADA* model by a technique called "Latent Walk".

The Latent Walk takes place in the Latent Space of the neural network. The generative process is closely linked to the Latent Space. It is through the training that the generator learns how to map points in the Latent Space and can depict specific output images. With each training the mapping is different and possess therefore a unique structure. Each time this structures is queried and navigated the results and the Latent Walk are different. To do a morph between the found images, we had to search the original dataset in the Latent Space, and figure out where the original data is represented in the Latent Space. To find the exact place of the original image we used a technique called Projection (See Figure 7 and Figure 8) to find the matching latent vector for each given image in the original dataset.

Figure 7: Target. Created by ART+COM Studios.

Figure 8: Projection. Created by ART+COM Studios.

After computing the latent vector for each image a Latent Walk (See Figure 9) between the found images was possible. First the images in the dataset needed to be morphed between each other, then we could generate new drawings of botanical illustrations.

Figure 9: Latent Walk. Created by ART+COM Studios.

3.6 Technical Setup

The technical setup for this experiment was quite elaborated: we ran the *UMAP* computation process on a machine with *Ubuntu 20.04 LTS*, on two *GTX 2080ti* (12 GB) with 64 GB RAM. This process with 500,000 images took us about 3 days, which is – compared to the ridge detection – an acceleration of the process. Reducing the image resolution by 50 % resulted in a speed-up of the training pro-

cess by roughly 1 day. The limit of the *UMAP* was at about 900,000 images, we can therefore see a hardware limitation at this point. Until this number of images "consumer-grade" hardware can be used and the tool is available to a wide range of users and not only to technical experts with a strong hardware. With more than 900,000 images the tool becomes available only to experts with the necessary technical equipment. With the current environment it took us about one day to integrate a new dataset of 3,000 images into the visualization of Image Garden via the *UMAP* algorithm. For the rest of the computation, including Ridge Detection and Colour Recognition, four days were necessary.

4 Show Cases and Evaluation

We have used various datasets to test the Image Garden processing pipeline, the clustering results and the interactive visualization: 2,600 plant drawings, 50,000 images from museum collections, 5,500 pictures of minerals from Wikidata and Wikipedia, 30,000 images from an intern image database of ART+COM, as well as around 15,000 images from *Bauhaus-Archiv, Museum für Gestaltung* in Berlin.

Figure 10 shows the representation of different datasets, that differ in the amount of images and topics, within a three or two dimensional space. The difference of the distribution in space and the forming of clusters is different in each of the dataset.

We also implemented prototypical, browser-based user interfaces for zooming, filtering, sorting, searching, maps, and timelines. These prototypically implemented user interfaces are used as a base for two developments currently under way: on the one hand, an interface for a museum exhibit which will allow visitors to navigate a large Image Garden projection; on the other hand a toolbox for curating digital collections, allowing knowledge workers not only to navigate content but also to add new content or modify existing one.

After computing the latent vector for each image of the original dataset, we where then able to morph between the images by hovering with the mouse cursor over the *UMAP* and compute a morphed animation.
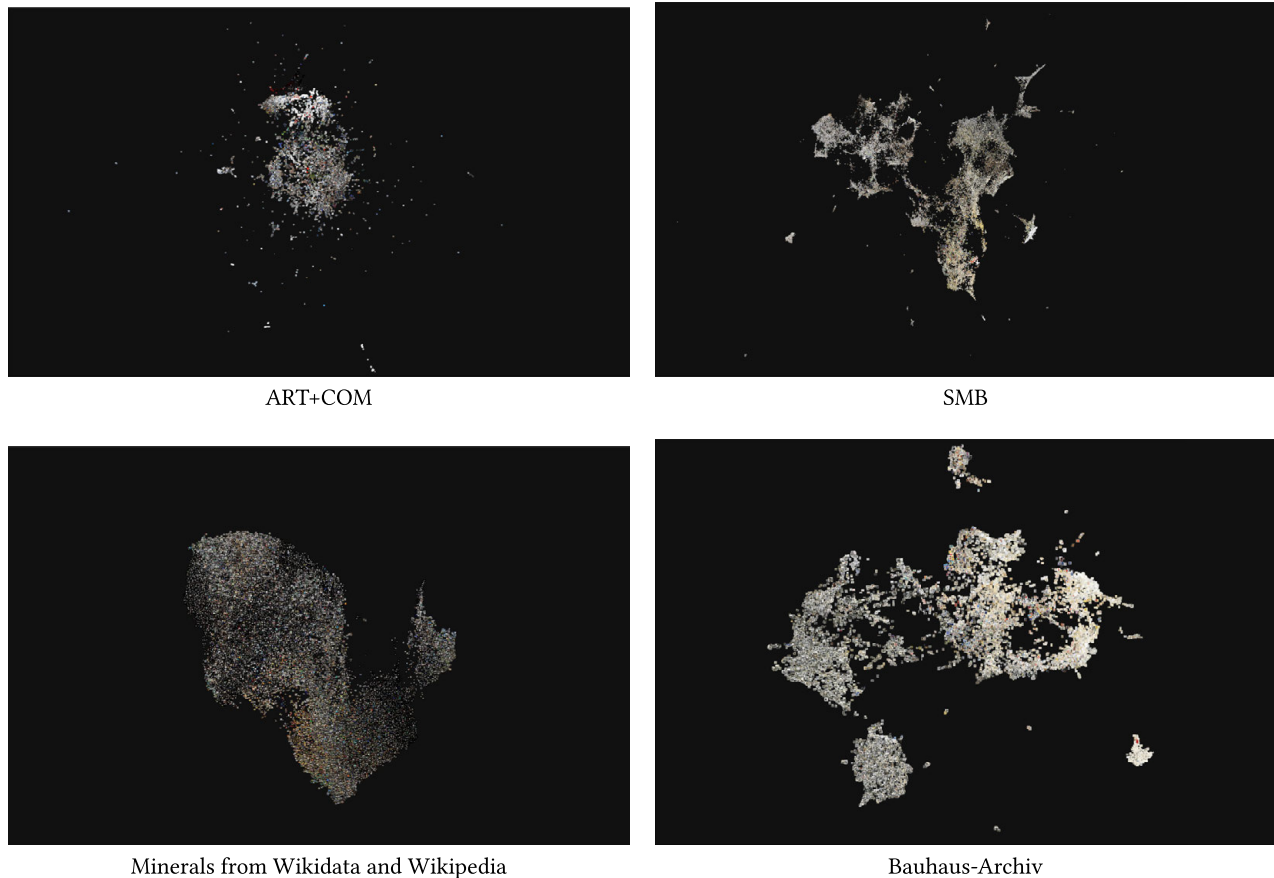

ART+COM

SMB

Minerals from Wikidata and Wikipedia

Bauhaus-Archiv

Figure 10: Visualization of several image collections. Created by ART+COM Studios.

4.1 Bauhaus-Archiv

Recently we had the chance to collaborate with the **Bauhaus-Archiv, Museum für Gestaltung** [2] for an installation in their Temporary Space in Berlin. The installation **Bauhaus Infinity Archive** [Bauhaus Infinity Archive. 2022. ART+COM Studios. https://artcom.de/en/ ?project=bauhaus-infinity-archive-2] (See Figure 11) has been opened in January 2022. The collection of *Bauhaus-Archiv* consist of over 1 Million different artefacts, only few where ever presented to the public. Over 16,000 of them were used to create **Bauhaus Infinity Archive**. The main concept of the installation is to give the visitor a vast and non-conventional impression of the archive, presenting a wide part of the image collection of the *Bauhaus-Archiv* in an immersive space.

Since drawing and the colour theory played a fundamental role in the school of *Bauhaus*, we create an interactive tool for the visitors to navigate in the archive by drawing a line and choosing a colour, making use of the technology from Image Garden. The software, which has been developed for the Back-End of the installation, showed satisfying results regarding lines and colours: a wide range of different forms and colours were present, moving away from the stereotype, that *Bauhaus* artworks contain only the three forms, square, circle and triangle as well as only the three colours, red, blue and yellow.

The collaboration with the *Bauhaus-Archiv* is an example how AI-based technologies can be integrated in an artistic context, embedding visual museum content in a three dimensional space, where the visitor can navigate in an intuitive way. With the help of the developed tool Image Garden users can interact with computers without the need of a deep understanding of the underlying technologies. They rest on a visual level of the collection being immersed in the three dimensional space.
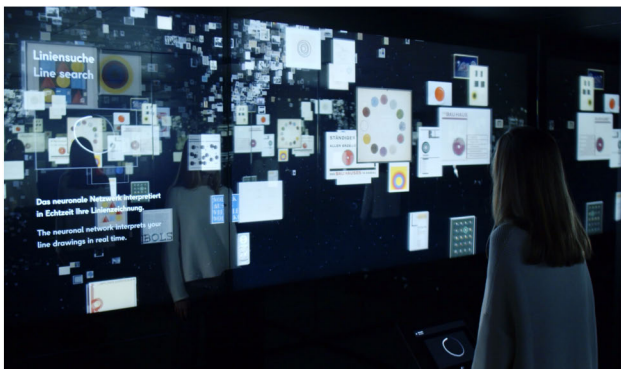

Figure 11: *Bauhaus Infinity Archive* in Berlin.

5 State of the Art

In the last years more and more tools to manage data with digital methods were developed. Many sectors use already different technologies to handle their data and ameliorate through this their work. In medicine we can find i. e. the PACS, a picture archiving and communication system that enables archiving data from diagnoses like x-rays or ophthalmological analyses. With the rise of tools that produce better images and digitalize content, the amount of data is growing. This growth will most likely continue in the next years as those technologies become more accessible and integrate themselves into the workflow of more and more areas.

Image clustering tools can also be found in the cultural sector were different approaches are used to cluster collections of diverse content. The tools behind those clustering methods vary according to their specific goals.

We want to take a closer look to existing tools and compare them to Image Garden. As a first example we want to look at **pixolution** [22]. It is a tool to explore an image database by looking for colours, similar images and text or other metadata. The results obtained show the images corresponding to the given search but the user does not have any link to the entire collection and does not see the rest of it.

Image Garden on the other hand is basically concepted as a tool to explore a wide collection of pictures, always maintaining a visual overview of the whole dataset, instead to being only a search engine. At the same time Image Garden offers the possibility to search for specific images from the collection based on a pure visual level (no text input), by drawing lines and free forms. This allows a new kind of search, that on one hand contains an intuitive artistic approach through the act of drawing the line by the user itself, and on the other hand could be the most intuitive way to explore pieces of arts, as in an installation, that we are currently creating for the *Bauhaus-Archiv, Museum of Gestaltung* in Berlin. The colour search of Image Garden is furthermore very sensitive and allows us to find a colour even if it is not very dominant in the image.

In our research our starting point was the program **PixPlot** [35], that we wanted to extend further. The program clusters images in a 2- or 3-dimensional visualization. Images that are similar can be found next to each other. This program was the base for further tools that we developed for this environment as the possibility of a search by lines and colours or the creation of a generative navigation through the Latent Walk inside the 3D map of the images.

The idea of searching for lines was to gain a dynamical interaction for the user with the collection of images. While exploring the dataset, unforeseen images would appear and the user would be surprised by the findings. Another approach of interactivity with the user can be found in the tool Fashion **MNIST** [20]. The art of exploring the database is very statical, since you can just navigate inside the 3D map of the fashion items. The aim of Image Garden was to create a more dynamical and playful search.

Looking further in the creative field the project **"Draw to Art"** [12] by Google Creative Lab, Google Arts & Culture Lab, IYOIYO is worth to be mentioned. The application uses machine learning to match sketches to drawings, paintings and sculptures from museums around the world. In the work users can make a hand-drawing on one side and similar artworks are shown on the other, making them accessible to museum visitors in a playful way.

The project used a deep neural network to recognize visual features in doodles and associated them to similar features in artworks. For the Doodle recognition the project uses the dataset of over 50 million drawings by the dataset **"Quick, Draw!"** [13] Once a sketch by the visitor is found in the *"Quick, Draw!"* dataset, the metadata with labels like basket, bird or nose – is compared with the metadata of the Google Arts & Culture image database and returns a matching image.

Compared to Image Garden the project *"Draw to Art"* uses two databases and compares them through its metadata to show similarities. This approach is quite different to Image Garden. Image Garden uses the same dataset for input and output in the project. We generate our own dataset by analyzing and paying attention to each individual line in an image dataset instead of finding shapes of faces, tigers or cars. This enables us a more precise result of found images, instead of just returning a result that contains general attributes like faces or basketballs as it is the case in the project *"Draw to Art"*.

6 Conclusion and Future Work

Image Garden integrates various AI technologies into a prototype for clustering and searching large image collections. Our focus has been on testing the capabilities of these technologies, integrating them into tools which can ultimately be used by knowledge workers for their work or by visitors in the exhibition of the museum. We also wanted to obtain general feedback concerning their usefulness in the context of curating collections and designing smart exhibitions.

The results obtained so far with respect to the processing pipeline are rather promising: usage of the technologies involved is straight-forward, the integration of a new dataset takes about one week, user interfaces with interactive visualization respond in real-time, and curators express great interest in the functionality provided. However, evaluating precision or accuracy of the clustering is notoriously difficult, as there is no established ground truth available. We will therefore focus on evaluations regarding, usability, user experience and benefits observed for knowledge workers and visitors.

Curators especially appreciate the fact that Image Garden provides an overview over image collections according to different search criteria e. g. on the visual level as well as the possibility to link their objects to real-world knowledge, such as Wikidata or other collection databases, as a basis for enhanced storytelling. The cluster information is also seen as valuable for completing and cleaning up metadata. This could be still further explored with the right dataset available.

Within a normal collection the visitor is not capable of grasping in a rational way the whole picture of the collection. The immersive experience enables a way to observe the variety and discover the connections without the need of using prior knowledge. For visitors Image Garden offers a way to discover the collection in a playful way and gives a possibility to grasp the specific images or items within their context. Another aspect, that could be further explored is the integration of personal inputs of the visitors and the link to the Image Garden Tool as for example specific experiences or physical appearances of the visitor.

We are already investigating to generate the *Annoy* Metric Tree with coloured lines and did some successful nearest neighbour tests within a random generated dataset. Furthermore we need to fully implement this step into our pipeline, extract the color for each line in the original dataset and build the *Annoy* Tree with it.

In the next step we would like to integrate the Image Garden pipeline with various other tools and technologies we have developed at ART+COM, namely entity browser, knowledge graph, contactless interaction and others. These extensions will be tested and evaluated in several show cases currently developed with partners and clients.

References

[1] Alexandr Andoni, Piotr Indyk, and Ilya Razenshteyn. 2018. Approximate Nearest Neighbor Search in High Dimensions. arXiv:cs.DS/1806.09823.

[2] 10625 Berlin Bauhaus-Archiv e. V. | Museum für Gestaltung, Schillerstraße 9. 2021. Bauhaus-Archiv. https://www.bauhaus.de/de/.

[3] Internationale Beleuchtungskommision. [n. d.]. CIELAB. http://www.cielab.de.

[4] Erik Bernhardsson. 2021. Ann benchmarks. https://github.com/erikbern/ann-benchmarks.

[5] QURATOR Bündnis. 2021. Qurator. https://qurator.ai.

[6] The SciPy community. 2021. K-Dimensional Tree. https://docs.scipy.org/doc/scipy/reference/generated/scipy.spatial.KDTree.html.

[7] 10559 Berlin DFKI GMBH, Alt-Moabit 91C. 2021. Digitale Kutarierung. https://artcom.de/en/.

[8] Piotr Dollaŕ and C. Lawrence Zitnick. [n. d.]. Structured Forests for Fast Edge Detection. *2013* ([n. d.]).

[9] Python Software Foundation. 2021. Annoy. https://pypi.org/project/annoy/1.0.3/.

[10] Python Software Foundation. 2021. Ridge Detection. https://pypi.org/project/ridge-detection/.

[11] Bundesministerium für Bildung und Forschung. 2021. BMBF. https://www.bmbf.de/bmbf/de/home/homenode.html.

[12] Google. 2021. "Draw to Art" by Google Creative Lab, Google Arts & Culture Lab. https://experiments.withgoogle.com/draw-to-art.

[13] Google. 2021. "Quick, Draw!" by Google Arts & Culture Lab. https://quickdraw.withgoogle.com/data.

[14] Jing He and Joachim Quantz. [n. d.]. Interactive knowledge visualization tools for exhibition curation. *International Journal of Computing*, Article 3 ([n. d.]), 173–170 pages.

[15] Keras. 2021. Keras. https://keras.io.

[16] Scikit learn developers. 2020. KMeans. https://scikit-learn.org/stable/modules/generated/sklearn.cluster.KMeans.html.

[17] McInnes Leland, Healy John, and Melville James. 2018. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. http://arxiv.org/abs/1802.03426.

[18] museum digital. 2021. SMB, *Staatliche Museen zu Berlin*. https://smb.museum-digital.de.

[19] Nanonets. 2021. Data Augmentation | How to use Deep Learning when you have Limited Data Part 2. https://nanonets.com/blog/data-augmentation-how-to-use-deep-learning-when-you-have-limited-data-part-2/.

[20] Inc. Observable. 2021. Fashion MNIST. https://observablehq.com/@stwind/exploring-fashion-mnist.

[21] François Pirsch. 2015. Vantage Point Trees. https://github.com/fpirsch/vptree.js.

[22] Weigandufer 45 12059 Berlin pixolution GmbH, c/o Büro 2.0. 2021. Pixolution. https://pixolution.org.

[23] NVIDIA Research Projects. 2021. StyleGAN2-Ada Github Repository. https://github.com/NVlabs/stylegan2-ada/.

[24] NVIDIA Research Projects. 2021. StyleGAN2 Github Repository. https://github.com/NVlabs/stylegan2.

[25] Leland McInnes Revision. 2018. UMAP. https://umap-learn.readthedocs.io/en/latest/.

[26] SmugMug+Flickr. 2021. flickr. https://www.flickr.com/people/biodivlibrary/.

[27] Princeton University Stanford Vision Lab, Stanford University. 2020. ImageNet. https://image-net.org.

[28] ART+COM Studios. 2021. ART+COM. https://artcom.de/en/.

[29] ART*COM Studios. 2021. NuForm. https://artcom.de/en/news/proximity-despite-distancing-new-forms-of-interpersonal-communication-in-museum-spaces-nuform/.

[30] Christian Szegedy, Wei Liu, Yangqing Jia, Sermanet Pierre, Reed Scott, Anguelov Dragomir, Erhan Dumitru, Vincent Vanhoucke, and Rabinovich Andrew. 2015. Going deeper with convolutions. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 1–9. https://doi.org/10.1109/CVPR.2015.7298594.

[31] Tensorflow. 2021. Tensorflow. https://www.tensorflow.org.

[32] Karras Tero, Laine Samuli, and Aila Timo. 2019. A Style-Based Generator Architecture for Generative Adversarial Networks. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 4396–4405. https://doi.org/10.1109/CVPR.2019.00453.

[33] Alexander von Humboldt. 2017. *Das graphische Gesamtwerk*. Lambert Schneider Verlag. 800 pages.

[34] Ross Wightman. 2016. ImageNet Script. https://github.com/rwightman/tensorflow-models/blob/master/tutorials/image/imagenet/classifyimage.py.

[35] Digital Humanities Laboratory Yale University Library. [n. d.]. PixPlot. https://dhlab.yale.edu/projects/pixplot/.

[36] Digital Humanities Laboratory Yale University Library. 2021. Yale Digital Humanities Lab Team. https://dhlab.yale.edu.

Bionotes

Eugenia Sinatti
ART+COM Studios, Berlin, Germany
eugenia.sinatti@artcom.de

Photo: Heimo Schulzer.

Eugenia Sinatti is software developer at the Research Department of ART+COM in Berlin. She received a Bachelor of Science in Mathematics from Technische Universität Berlin, Germany. Her current field of work is focused on the application of machine learning processes in museums and exhibitions, combining quantitive and qualitative aspects of computer vision.

Simon Weckert
ART+COM Studios, Berlin, Germany
sw@simonweckert.com

Photo: Raphael Wild.

Simon Weckert is artist and designer at the research department of ART+COM in Berlin. He graduated from the University of the Arts Berlin, Germany. In his works he shares knowledge from generative design to physical computing. His focus lies in the digital world – everything linked to code and electronics with a critical view on current social aspects, ranging from technology oriented examinations to the discussion of current social issues.

Ewelina Dobrzalski
ART+COM Studios, Berlin, Germany
ewelina.dobrzalski@artcom.de

Photo: Tomek Pietrzyk.

Ewelina Dobrzalski is research coordinator at ART+COM in Berlin. She worked for the University of the Arts Berlin, the Goethe-Institut Kraków and other cultural institutions. Passionate about the in-between spaces of disciplines she is also member of the advisory board of "The shape of things to come". The project explores the potential of performing arts as a catalyst for scientific inquiry and platform for new human-machine interactions.