Rijksmuseum
Data system project

Conor O'Donnell, Joran Timmerman, Min Yi Zhang, Noemi Roos, Wenjia Gu

TABLE OF CONTENTS

01

Problem definition

Current Rijksmuseum Search Engine

Boasts a large collection with purely textual search functionalities

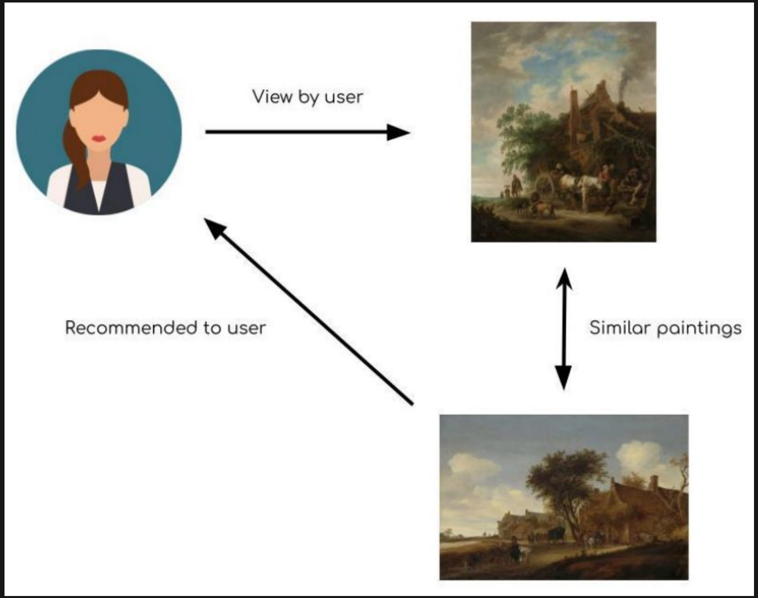
Search functionalities don't prioritize discovery for certain visual features

What we aim to improve

Implement a search functionality that combines existing textual metadata and visual elements

Guide the user into discovering pieces

Summarizing the large dataset in a personalized, digestible and appealing way



02

Solution

System Components

User feedback

Track user clicks and refreshes to incorporate better results on the homepage

Target Features

1. Art Style/Time Period
2. Colour
3. Visual Embeddings (CLIP)
4. Textual Description

Recommender System

Make five painting recommendations at a time. Four correspond to the four target features, and one is a combination of all features.

Recommendations will be similar to the reference painting but with some degree of difference to avoid filter bubbles.

Colour Feature

- Get the dominant colour from each painting (highest percentage)
- Convert hex colour string to the corresponding Lab colour space
- Calculate the Euclidean distance between two Lab values and create a similarity matrix

→ The smaller the Euclidean distance between two colours, the more similar the colours are

→ Lab preferred over RGB because it is more perceptually uniform → difference between colours are more closely related to the human perception of colour

$$d(p,q) = \sqrt{(L_p - L_q)^2 + (a_p - a_q)^2 + (b_p - b_q)^2} \quad (1)$$

Where:

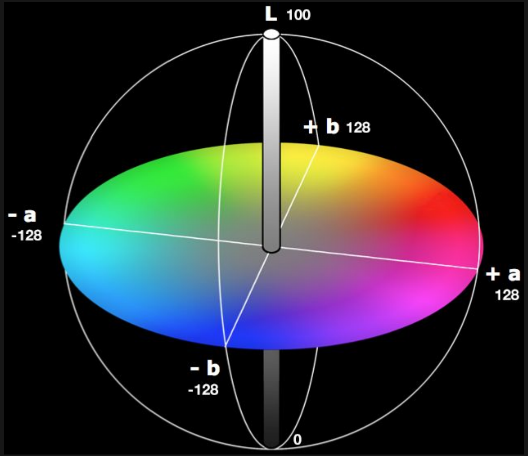
$$d = \text{Euclidean distance} \quad (2)$$

$$p, q = \text{lab values} \quad (3)$$

$$L_p, L_q = \text{Lightness values of the two lab values} \quad (4)$$

$$a_p, a_q = \text{a-channel values of the two lab values} \quad (5)$$

$$b_p, b_q = \text{b-channel values of the two lab values} \quad (6)$$



CLIP visual embedding

- Zero-shot learning capabilities eliminated the need for further training
- CLIP ViT-B/32 visual transformer was used to embed images
- Images were preprocessed to conform with the size the model expects
- We used Cosine similarity as the similarity measure after normalizing and computing the dot product

- Not purely visual, but also contextual
- Recognition of similar concepts

“Astronaut” more similar to “Rocket” than it is to “Man”

Text Description Feature

- Create a list of stop words that will be removed from the descriptions before vectorizing them
- Apply TfidfVectorizer to the text descriptions of paintings and create a sparse matrix of the TF-IDF values for each description
- Use the linear_kernel method to compute the the dot product of the TF-IDF matrix with itself and create a similarity matrix

→ Similarity score: measure of how similar two descriptions are to each other

The Night Watch, Rembrandt van Rijn, 1642

oil on canvas, h 379.5cm × w 453.5cm × w 337kg More details

Rembrandt's largest and most famous painting was made for one of the three headquarters of Amsterdam's civic guard. These groups of civilian soldiers defended the city from attack. Rembrandt was the first to paint all of the figures in a civic guard piece in action. The captain, dressed in black, gives the order to march out. The guardsmen are getting into formation. Rembrandt used the light to focus on particular details, like the captain's gesturing hand and the young girl in the foreground. She was the company mascot. The nickname Night Watch originated much later, when the painting was thought to represent a nocturnal scene.

On display in Nightwatch gallery

Visit via the The best of the Rijksmuseum tour

Visit via the Rembrandt tour

Art style/Time period

We used ResNet-18 pre-trained model and fine tuned it with the ArtGAN Dataset to create an Art Style Classifier. Then we used it to classify each painting into one of 27 different art style classes.

We mapped out a time period for each Art Style based on information available online

Then we were able to create a similarity matrix using Euclidean Distances between all time periods to get a value for "similarity" between different art styles.

Style	Start	End
Early Renaissance	1401	1495
Northern Renaissance	1430	1580
High Renaissance	1490	1527
Mannerism Late Renaissance	1520	1600
Ukiyo-e	1603	1867
Baroque	1600	1730
Rococo	1720	1760
Romanticism	1780	1850
Realism	1830	1870
Impressionism	1860	1890
Pointillism	1886	1900
Post Impressionism	1886	1905
Symbolism	1880	1910

Art Nouveau	1890	1914
Fauvism	1904	1909
Naive Art (Primitivism)	1885	1960
Analytical Cubism	1907	1914
Synthetic Cubism	1907	1914
Cubism	1907	1914
Expressionism	1905	1930
Abstract Expressionism	1943	1960
Action Painting	1940	1960
Color Field Painting	1947	1960
Pop Art	1955	1965
Minimalism	1959	1975
New Realism	1960	1962
Contemporary Realism	1960	1975

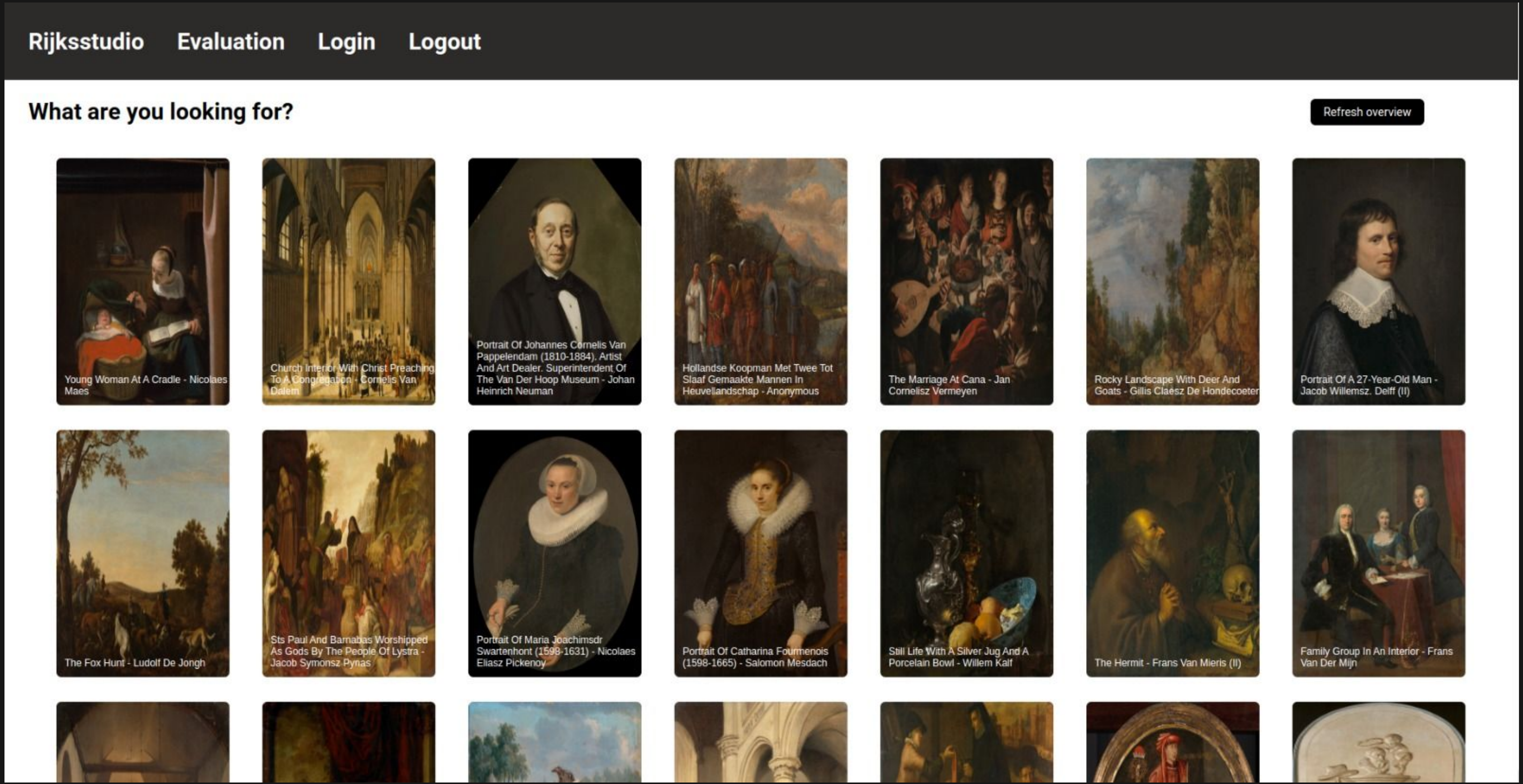
```
euclidean_distance =
np.sqrt((start1 - start2)**2 +
(end1 - end2)**2)
```

Combination of Features

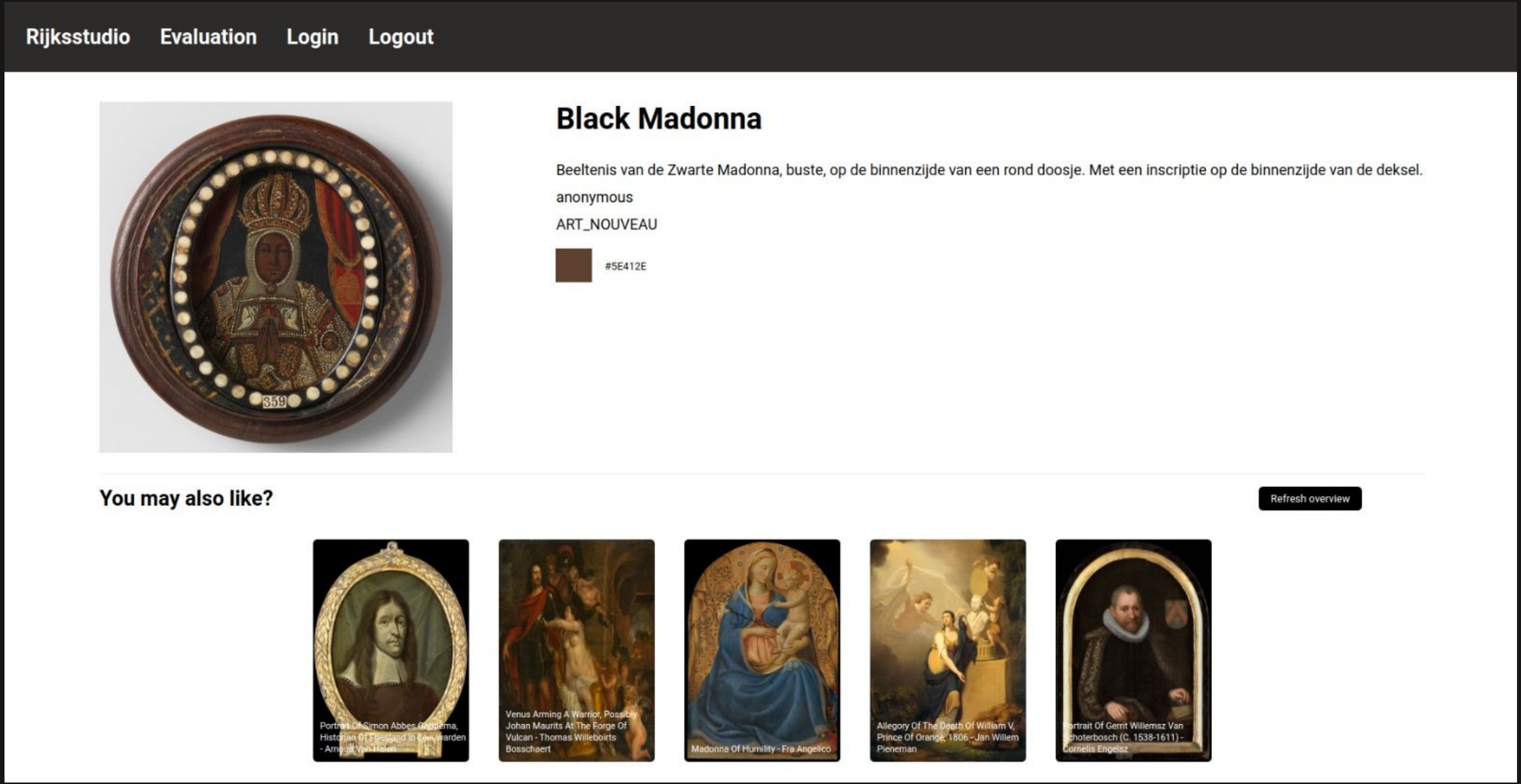
- The final recommendation suggested to the user is based on an equal weighting combination of the four target features.

- The recommendation displayed will be the painting with the highest similarity score from this calculation.

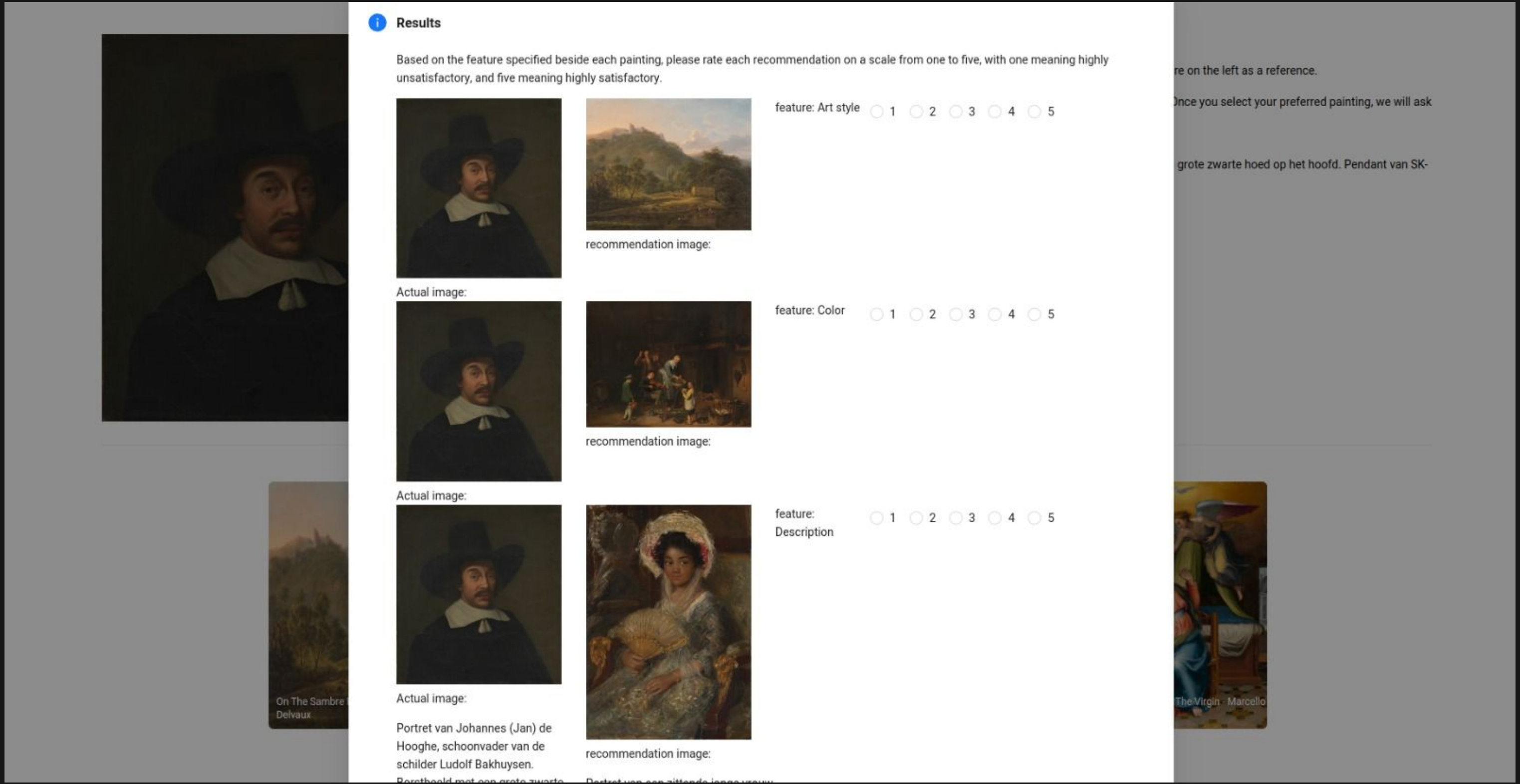
User Interface – Homepage

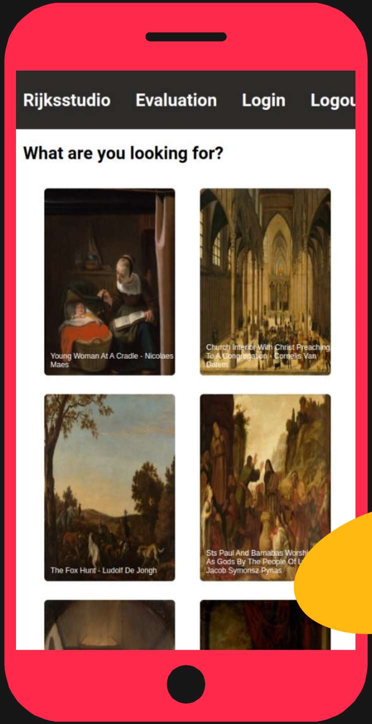


User Interface – Individual Painting Page



User Interface – Evaluation Page





Demo

← short video demonstration→

03

Validation

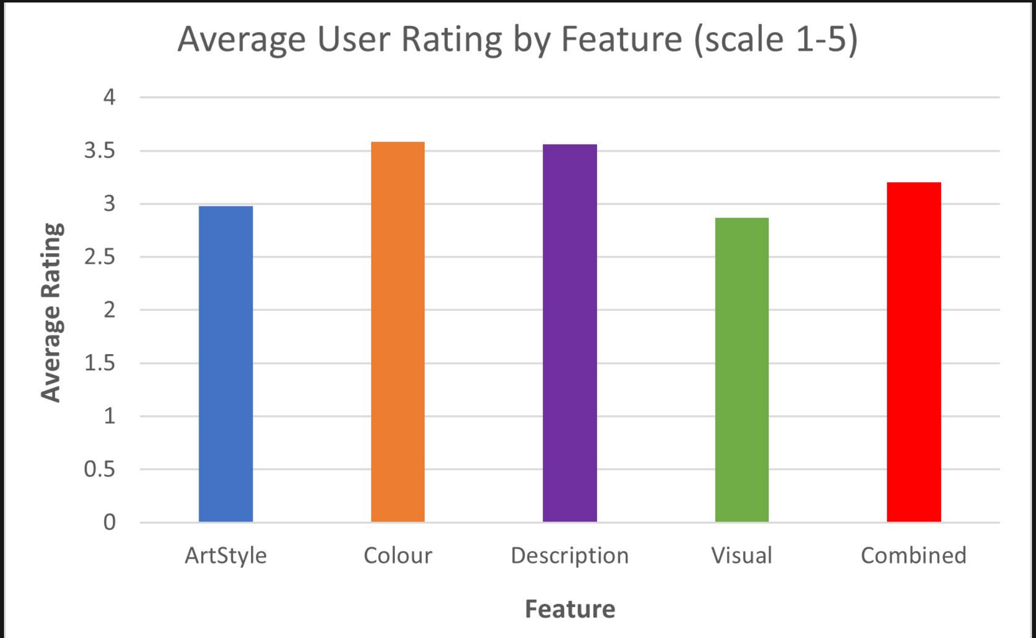
Validation: User Behaviour Analysis

- We want to to compare user behaviour in the application itself with user behaviour in the evaluation interface.
- Allow the user to engage with the application for a specified period and track their clicks.
- Likewise, track their clicks in the evaluation interface.
- Then we can compare if user behaviour remains the same once they become aware of the features we are using and how our recommendations are structured.
- We have not yet completed this analysis. We have been tracking clicks but not enough validation data yet.

Validation: Feature Ratings

- We would like to evaluate how users feel about the strength or validity of each recommendation feature.
- Users are presented with 5 recommendations, each one corresponding to one of our features (or the combination of all) and asked to give each a score from 1 to 5.
- We can use this information to identify which features are underperforming, and maybe use it to decide the best weighting for our combination of all features.

Validation Results - Feature Rating



- As expected, users were most satisfied with the Colour and Textual Description Features.
- The CLIP visual embeddings were the least satisfactory feature.
- The Art Style feature and the Combination were moderately satisfactory.
- We will increase sample size of users evaluated before drawing more significant conclusions.

THANK YOU!

Conor O'Donnell, Joran Timmerman, Min Yi Zhang, Noemi Roos, Wenjia Gu