Vision, Modeling, and Visualization (2024), pp. 1–6
L. Linsen and J. Thies (Eds.)

Musicon: Glyph-Based Design for Music Visualization and Retrieval Supplementary Material

Xuejiao Luo, Vera Hoveling and Elmar Eisemann

Delft University of Technology, The Netherlands

1. Introduction

Given the subjective nature of visualizing and perceiving music, our hypothesis and experimental design are evaluated through user tests. The major focus of the evaluation lies in our proposed features: from the effectiveness of the icon to a larger system-evaluation.

The interface was implemented in a web app to facilitate remote user testing. It allows for real-time search in a database of 10K songs.

We targeted a participant demographic of “non-expert but generally computer-literate” adults and emphasized diversity in gender and age across ranges from 20-29 to over 70 years. To reduce response bias, participation was anonymous. Using an a-priori sample size calculator with an expected medium effect size ($d = 0.5$), we determined that a minimum of 27 participants would achieve a statistical power of 0.8 and a significance level of 0.05, assuming analysis via paired samples t-test for certain tasks. Consequently, we garnered 38 responses in the evaluation. The distributions for their respective age, gender and experience with the Spotify streaming service can be seen in Figure 1.

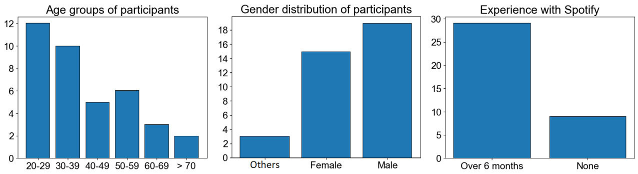


Figure 1: *Demographic information on the participants of the user study.*

The user study consists of five tests and each of them is based on the following tasks:

1. Test one: Visual clustering (free-grouping task).
2. Test two: Outlier detection (five-alternative forced-choice task).
3. Test three: Generalization, Contrast and CVD Robustness (matching-to-sample task).
4. Test four: Search-by-icon (real-world task).
5. Test five: Search-in-playlist (real-world task)

2. Test one: Visual clustering

2.1. Task

The goal of this test was to evaluate how well the icons capture the similarity of their features and how much users agree on this. Participants were asked to visually form clusters from a set of 60 icons, without knowing song titles or other information. Participants could use any number of clusters and were allowed to leave a set of spare icons that did not fit to anything else. An example screenshot for this test is shown in Figure 2. To ensure that there is a diversity in the selection yet still the possibility to make clusters, we sampled 10 data points from six of the clusters grouped by applying the k-means clustering algorithm ($k = 10$). Each participant worked with the same set of icons but their presentation was in a random order.

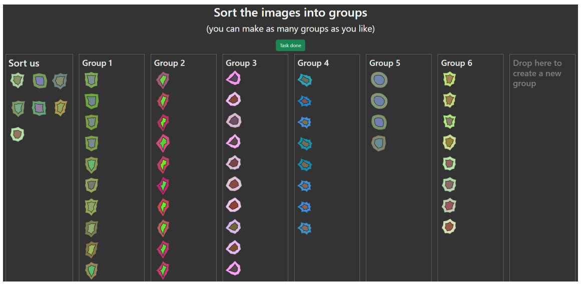

Sort the images into groups
(you can make as many groups as you like)

Figure 2: *An example screenshot for test one.*

2.2. Results and discussion

To see how users agree on the clustering, we calculated a co-occurrence matrix of the clusters made by participants and a cosine similarity matrix of the feature vectors. The resulting matrices can be seen in Figure 3.

Our analysis indicates a consensus among users regarding the clusters. Initial examination reveals a striking resemblance between the co-occurrence matrix and the similarity matrix. To quantify this

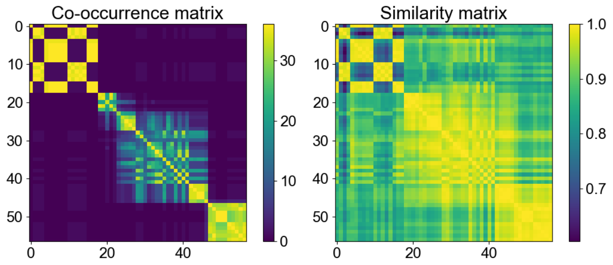


Figure 3: *Co-occurrence matrix of the clusters made by participants (left) and a cosine similarity matrix of the feature vectors (right).*

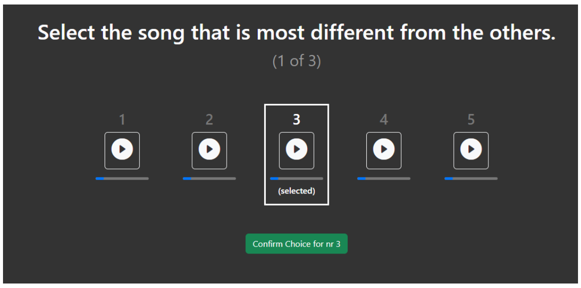
relationship, we computed the pairwise Pearson correlation coefficient, which resulted in a value of 0.6. This value denotes a 'moderate' linear correlation, suggesting a reasonable degree of agreement among users in their clustering decisions.

3. Test two: Outlier detection

3.1. Task

This test builds upon test one, by using the cluster data the participants provided themselves. It is in essence a five-alternative forced-choice task and is to see how well the icons represent similar music and if the clustering allows users to spot outlier songs easily.

For each participant, we randomly selected four songs from a single cluster, along with one song from a different cluster, and then presented these five songs in a randomized sequence. Participants were then asked to identify the song that sounded distinct from the rest. An example screenshot for this test is shown in Figure 4. We repeated this three times for each participant.



Select the song that is most different from the others.

(1 of 3)

Figure 4: *An example screenshot for test two.*

3.2. Results and discussion

The outlier recognition rates achieved by the participants can be seen in Figure 5 and the descriptive statistics of the results can be found in Table 1.

The expected recognition rate when of random guessing would be 0.2. It seems rather likely that our results with a mean recognition rate of 0.745 is a considerable improvement. We observe that p = 0.000000000000002 (one sample one-tailed t-test), finding an effect size of 2.375 (Cohen's d).

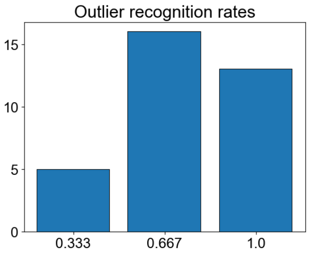


Figure 5: *Recognition rates obtained for outlier detection in test two.*

Mean	Median	Mode	Std	Variance
0.745	0.667	0.667	0.230	0.053

Table 1: *Descriptive statistics of the recognition rates obtained for outlier detection.*

4. Test three: Generalization, Contrast and CVD Robustness

4.1. Task

The goal of this test was threefold:

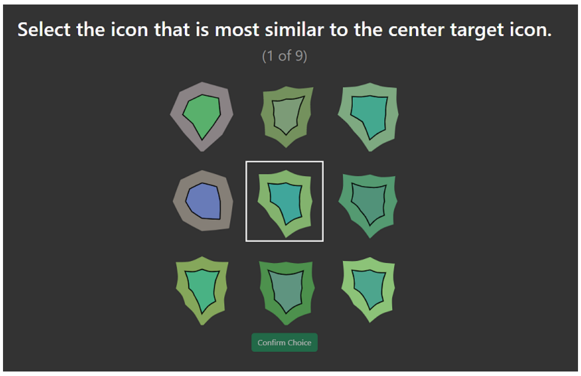
1. Evaluate if the most similar icon aligns with the data point having the highest cosine similarity. In other words: if the icon is generally meaningful for representing high-dimensional data.
2. Evaluate if the contrast enhanced version of the icon improves performance in terms of time-on-task and accuracy for finding the most similar icon.
3. Evaluate the robustness of the icon design against colour blindness by testing the time-on-task and accuracy with a color vision deficiency-simulated version of the icon.

We presented the user with nine icons that are all rather similar. One of the icons was the target icon. We asked participants to select the icon most similar to the target icon. An example screenshot can be seen in Figure 6.

We performed this test for three different 'rendering modes':

- 'default', as the icon was designed and explained in Section 3.
- 'contrast', with 100% increase of contrast between the nine icons, as explained in Section 4.4.
- 'CVD', with CVD simulated on the colour rendering, more specifically deuteranomaly - the most common form of colour blindness

We repeated the task nine times for each participant: three times for each rendering mode.



Select the icon that is most similar to the center target icon.

(1 of 9)

Figure 6: *An example screenshot for Test 3.*

4.2. Results and discussion

Overall, we find that the icon performs well in the matching-to-sample task. A one-way ANOVA test conducted across all three rendering modes yielded a p-value of 0.450, indicating that rendering mode has no noticeable impact on performance in the matching-to-sample task. This outcome suggests that each icon rendering mode performs comparably well, affirming the robustness of our icon to color blindness. The effectiveness of our redundant encoding strategy in enhancing recognition and matching accuracy is thus supported by these results.

4.2.1. Comparison between 'default' and 'contrast' icons.

Figure 7 and Table 2 present a comparison of recognition rates between the high contrast and 'default' rendering of the icon. Similarly, Figure 8 and Table 3 display a comparison of time-on-task between the high contrast and 'default' rendering of the icon.

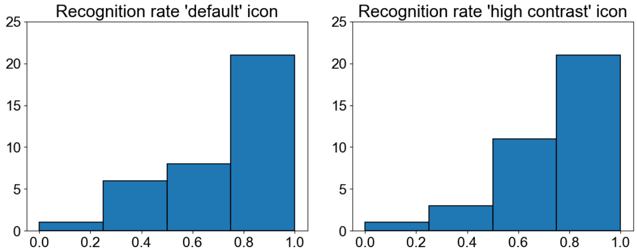


Figure 7: *Comparison of recognition rates obtained for matching-to-sample with the 'default' and 'contrast' version of the icon.*

Mode	Mean	Median	Mode	Std	Variance
Default	0.706	0.750	1.000	0.277	0.077
Contrast	0.785	1.000	1.000	0.271	0.074

Table 2: *Descriptive statistics of the data as displayed in Figure 7.*

We had expected the high contrast version to yield higher recognition rates than the default version. Inspecting the plots and the statistical descriptions, this seems to be the case: From Table 2, we notice that for high-contrast rendering, the mean recognition rate is higher than for the 'default' version of icon and indeed we see the distribution in Figure 7 shift a bit to the right. However, for a statistical analysis, we find a medium-sized effect size of 0.283 (Cohen's d), meaning that there is no sufficient statistical significance (p = 0.103 for a one-tailed paired-samples t-test).

In terms of time-on-task, we had expected the high contrast icon to allow for faster selection than the default icon and that seems to be indeed the case. As with the recognition rates, we find a medium-sized effect of 0.202 (Cohen's d) but fail to establish strong significance: p=0.084 (one-tailed paired-samples t-test).

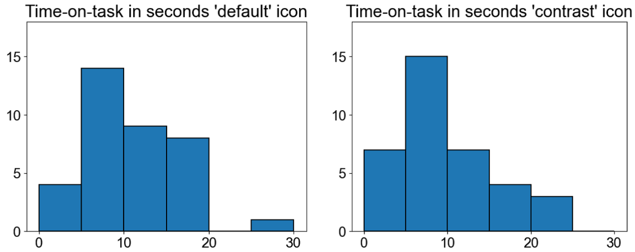


Figure 8: *Comparison of the time-on-task for matching-to-sample with the 'default' and 'contrast' version of the icon.*

Mode	Mean	Median	Mode	Std
Default	10.862	9.742	5.446	29.658
Contrast	9.754	7.812	5.385	28.999

Table 3: *Descriptive statistics of the data as displayed in Figure 8.*

4.2.2. CVD robustness

We are interested in a comparison of a CVD simulated mode icon with the 'default' icon, to see if the redundant encoding in our icon indeed makes the icon more robust to CVD. Therefore we compare the recognition rates we found in the sample matching for the 'default' and CVD rendering modes. In Figure Figure 9, we can compare the distribution of the recognition rates participants achieved for the high contrast version of the icon with the default rendering of the icon, Table Table 4 provides the descriptive statistics for the data.

Up front, we hypothesised that the CVD version would underperform slightly in comparison with the default version of the icon. There seems to be a change in the distribution, where the median value does shift from 0.750 to 0.667 (Table 4. We find that the mean recognition is a bit higher but this might be statistical noise, as we cannot confirm any statistical significance between these distributions: a two-tailed paired-samples t-test yields a p value of 0.715.

Mode	Mean	Median	Mode	Std	Variance
Default	0.706	0.750	1.000	0.277	0.077
CVD	0.741	0.667	0.667	0.231	0.053

Table 4: *Descriptive statistics of the data as displayed in Figure 7.*

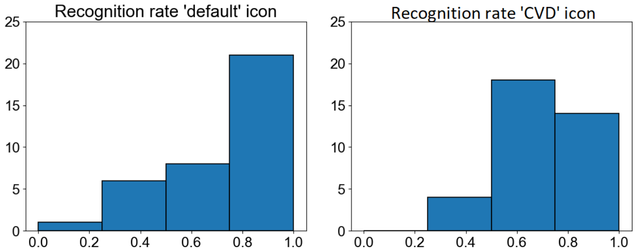
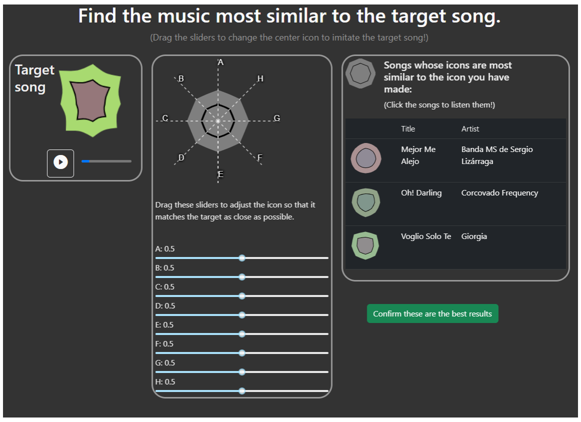


Figure 9: *Comparison of recognition rates obtained for matching-to-sample with the 'default' and 'CVD' version of the icon.*

5. Test four: Search-by-icon

5.1. Task

The test aimed to assess the efficacy of our 'search-by-icon' method for users. Participants were shown a target song with its custom icon and the search-by-icon interface shown in Figure 10. They were tasked with using the interface to imitate the target icon and then retrieve the three songs most similar to the target one.


Find the music most similar to the target song.

(Drag the sliders to change the center icon to imitate the target song!)

Drag these sliders to adjust the icon so that it matches the target as close as possible.

A: 0.5
B: 0.5
C: 0.5
D: 0.5
E: 0.5
F: 0.5
G: 0.5
H: 0.5

Songs whose icons are most similar to the icon you have made:

(Click the songs to listen them!)

	Title	Artist
	Mejor Me Alejo	Banda MS de Sergio Lizárraga
	Oh! Darling	Corcovado Frequency
	Voglio Solo Te	Giorgia

Figure 10: *An example screenshot for Test 4.*

5.2. Results and discussion

We evaluate this novel method by evaluating the different sub-tasks we distinguished: how close the user can imitate an icon, how well the user can retrieve 'similar' music with this tool, and the willingness of users to adopt this tool with the System Usability Scale.

5.2.1. Imitated icon and retrieved songs

Cosine similarities between user-generated and target icon vectors, presented in Figure 11 (left), with a high mean (0.989) and median (0.993), indicates that with vectors exhibiting a cosine similarity above 0.975 to the target, most users accurately replicated icons. Furthermore, the average cosine similarities between the target icon vector and the top three selected songs, detailed in Figure 11 (right), reinforce the precision of these imitations, highlighting the effectiveness of participant selections in aligning closely with the target icons. The descriptive statistics of the results can be seen in Table 5.

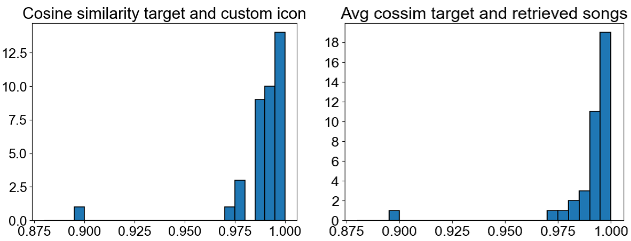


Figure 11: *Cosine similarities between user-generated and target icon vectors (left) and the average cosine similarities between the target icon vector and the top three selected songs (right).*

Task	Mean	Median	Std	Variance
Imitation	0.989	0.993	0.017	0.00003
Retrieval	0.991	0.995	0.017	0.00003

Table 5: *Descriptive statistics of the data from imitated icon task and retrieved songs task as displayed in Figure 11.*

5.2.2. System Usability Scale (SUS)

The SUS consists of the ten questions:

1. I think I would like to use this product frequently.
2. I found it unnecessarily complicated.
3. I found the product easy to use.
4. I think I need technical support to use the product.
5. I found the different functions of the product well integrated with each other.
6. I felt there were too many contradictions in the product.
7. I can imagine that most people can quickly get to grips with the product.
8. I found the product cumbersome to use.
9. I felt confident while using the product.
10. I had to learn a lot about the product before I could use it properly.

Each of these statements was ranked with the Likert Scale anchored with one for 'fully disagree' and five for 'fully agree'. The answers that were given in response to each of the questions in the SUS can be seen in Figure 12.

Based on the feedback, we computed the SUS scores, as illustrated in Figure 13(left), with the corresponding performance interpretations presented in Figure 13 (right). We recognize that condensing the user experience into a singular score can simplify the nuanced nature of their feedback. Nonetheless, we observe that a majority, specifically 25 out of 38 participants, demonstrates a willingness to embrace our model, despite the fact that it currently does not integrate the standard features derived from meta data, which we would not want to avoid in a final system but excluded to examine the effectiveness of our core contribution.

6. Test five: Search-in-playlist

6.1. Task

This test aimed to assess the icon's effectiveness and sorting properties within a playlist context, comparing it against the prevalent

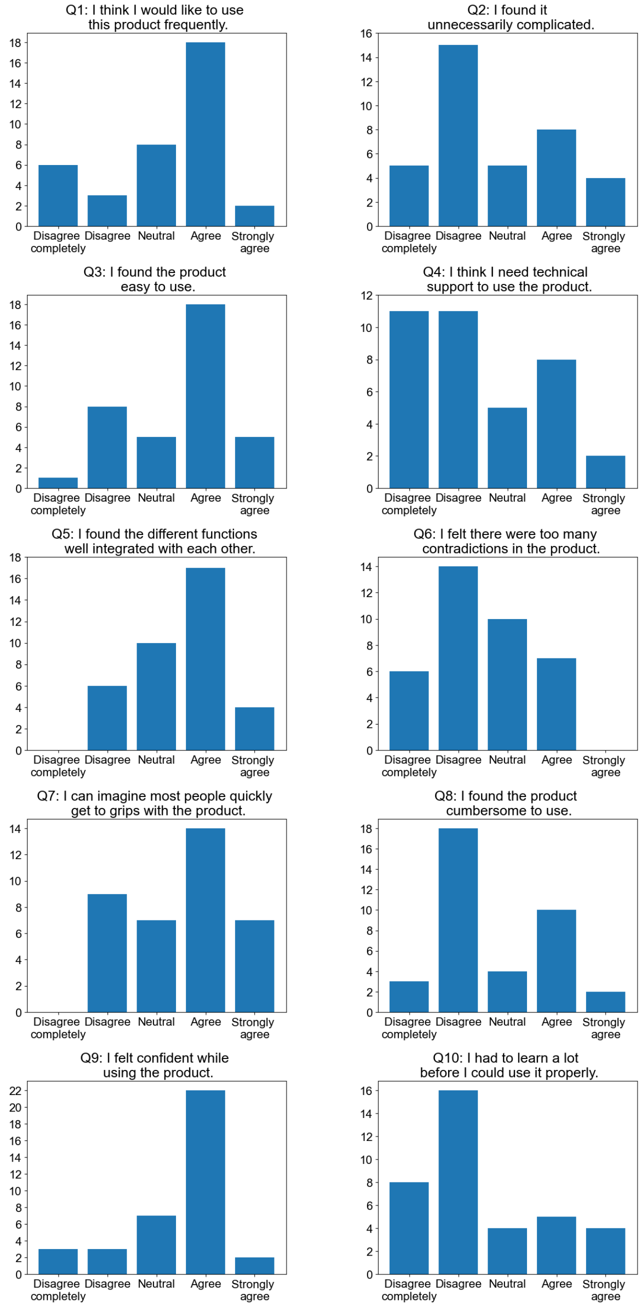


Figure 12: *Answers given to questions the SUS.*

use of album art in streaming services. Participants were asked to select their top three songs from playlists featuring both album art and our custom design, with each format presented twice. An example screenshot is shown in Figure 14. To prevent order effect, the target song was selected randomly from the selection of possible target songs, the playlist order was randomised for each participant, as was the order in which they were presented with custom icon and album art icons.

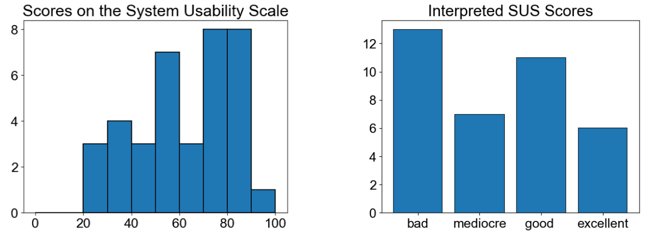
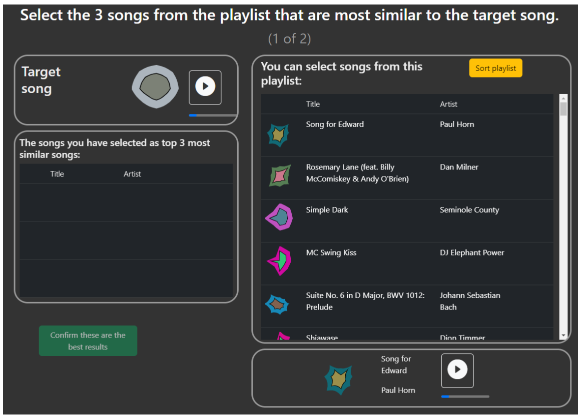


Figure 13: *SUS Scores (left) and its corresponding interpretation (right). A score above 80.3 is interpreted as 'excellent', scores between 68 and 80.3 as 'good', scores between 50 and 68 'mediocre', and anything below 50 'bad'.*



Select the 3 songs from the playlist that are most similar to the target song.

(1 of 2)

Target song

The songs you have selected as top 3 most similar songs:

Title	Artist

You can select songs from this playlist:

Title	Artist
Song for Edward	Paul Horn
Rosemary Lane (feat. Billy McComiskey & Andy O'Brien)	Dan Milner
Simple Dark	Seminole County
MC Swing Kiss	DJ Elephant Power
Suite No. 6 in D Major, BWV 1012: Prelude	Johann Sebastian Bach
Shiawase	Dion Timmer

Song for Edward

Paul Horn

Figure 14: *An example screenshot for Test 5.*

6.2. Results and discussion

We assessed the similarity between the top three selections and the target vector, time-on-task, plays per task, and additional insights from open-ended questions.

6.2.1. Retrieved songs

Average cosine similarities between the target icon vector and the top three selected songs are presented in Figure 15, with descriptive statistics in Table 1. Both methods cover similar ranges of cosine similarities, but the icon method facilitates slightly higher similarity retrieval (one-tailed paired-samples t-test: p = 0.03, Cohen's d: 0.469), aligning with the icon's intended similarity representation.

Icon	Mean	Median	Std	Var	Min	Max
Album	0.955	0.967	0.033	0.001	0.858	0.991
Custom	0.938	0.945	0.036	0.001	0.851	0.993

Table 6: *Descriptive statistics of the data as displayed in Figure 15.*

6.2.2. Time-on-task

The time-on-task per participant for both album art and custom icon methods are detailed in Figure 16, with descriptive statistics provided in Table 7. A notable reduction in average completion time,

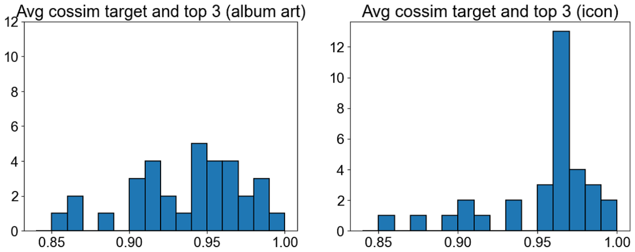


Figure 15: *Average cosine similarities between the target icon vector and the top three selected songs, comparing between album art (left) and our custom icon (right).*

exceeding one minute, was observed. A left-tailed paired t-test confirmed these findings with p = 0.00201 and an effect size of 0.473 (Cohen's d).

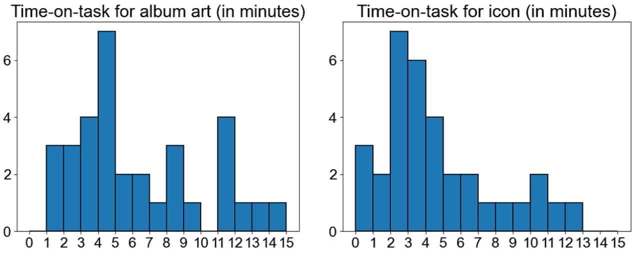


Figure 16: *Completion times of time-on-task with album art (left) and our custom icon (right).*

Icon	Mean	Median	Std	Min	Max
Album	6:25	4:49	3:45	1:08	14:12
Custom	4:44	3:54	3:11	0:39	12:14

Table 7: *Descriptive statistics of the data as displayed in Figure 16, formatted as mm:ss.*

6.2.3. Songs played per task

Figure 17 and Table 8 display the number of songs played per task per participant for both album art and the custom icon, showing similar ranges but a notably lower mean and median for the custom icon. A paired t-test confirms this difference, with p = 0.00001 and an effect size of 0.931 (Cohen's d).

Icon	Mean	Median	Std	Var	Min	Max
Album	103.9	113.5	55.9	3121.1	10	222
Custom	57.7	40.0	42.4	1796.8	7	192

Table 8: *Descriptive statistics of the data as displayed in Figure 17.*

6.2.4. Open Questions

Upon study completion, participants responded to three open-ended questions regarding their playlist task experience. We summarise and highlight the responses here.

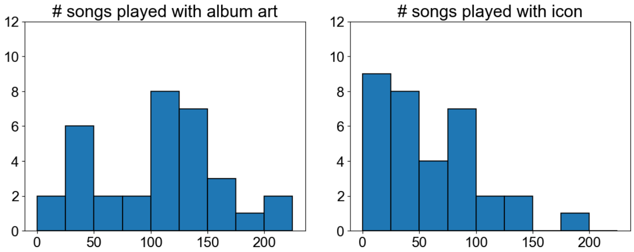


Figure 17: *The number of songs played per task per participant for both album art (left) and the custom icon (right).*

Q1: How did you experience using the custom icon differ from the regular setting?

To this first, question 23 participants mentioned a positive experience. 14 mentioned explicitly that they experienced that it made their task of music selection easier. Three participants mentioned that they considered the album art icon more fit for the task, of which two explicitly mentioned that they found the album art to give more information about the song than the custom icon.

Q2: Was there anything surprising or unexpected?

To this second question, five participants mentioned that they were surprised how well the icon had supported their task. In contrast five participants mentioned they had encountered, what they considered outliers, which made them doubtful in how well the icon reflected the music content in some cases.

Q3: What could be done to improve the icon?

To the third and final question we got some very concrete feedback from participants: four participants mentioned that they would like the icon to be more expressive, in terms of shape and colour. In particular, three mentioned the colours as a bit bland. In addition, 12 participants expressed a longing for a better understanding of the parameters of the icons: something more semantic or a more elaborate explanation, several of them mentioned a desire for genres mapped to axis or colours.

Overall, our tool's effectiveness was confirmed through strong quantitative results, notably speeding up task completion by over a minute compared to album art presentations and reducing the number of songs participants needed to listen to by almost 50%. When using our icon, selections tended to have similar or slightly higher cosine similarity to the target song, suggesting the icons' visual cues enhanced both the speed and quality of decision-making. While many participants valued our icon for its capacity to indicate similarity, three expressed a preference for album art due to its contextual and cultural insights. Acknowledging album art's value in certain situations, we argue that our icon meaningfully enhances user experience by addressing the variability of songs within an album.