Europeana DSI 2– Access to Digital Resources of European Heritage

DELIVERABLE

D6.1: Advanced image discovery report

Revision	Version 1.0
Date of submission	
Author(s)	Sergiu Gordea, AIT - Austrian Institute of Technology David Haskiya, Europeana Foundation Ash Marriott, Europeana Foundation
Dissemination Level	Public

Co-financed by the European Union
Connecting Europe Facility

REVISION HISTORY AND STATEMENT OF ORIGINALITY

Revision History

Revision No.	Date	Author	Organisation	Description
1	15.08.2017	Sergiu Gordea	AIT - Austrian Institute of Technology	Draft document
2	28.08.2017	Antoine Isaac, Timothy Hill	Europeana Foundation	Document Reviews
3	31.08.2017	Sergiu Gordea	AIT - Austrian Institute of Technology	Final version

Statement of originality:

This deliverable contains original unpublished work except where clearly indicated otherwise. Acknowledgement of previously published material and of the work of others has been made through appropriate citation, quotation or both.

The sole responsibility of this publication lies with the author. The European Union is not responsible for any use that may be made of the information contained therein.

‘Europeana DSI is co-financed by the European Union's Connecting Europe Facility’

Table of Contents

Executive Summary

This Deliverable presents the results from the work carried out within the scope of *Subtask 6.3.4 Develop image discovery services*. This deliverable is based on the milestone document MS6.1: Advanced image discovery development plan[1] (MS6.1), which presents the initial planning for the activities related to the development of advance image similarity search services. The detailed description of the development related activities and associated service documentation is presented in the following sections of this deliverable.

The related project subtask has the following formal description in Description Of Action document:

“Subtask 6.3.4. Develop content-based discovery services (infrastructure, search index, APIs) to support image search by similarity and to support improving and extending the existing browse by colour functionality. This will support not only image based discovery and browse in Europeana Collections but will also be made available for 3rd party use via APIs documented on Europeana Labs.”

We present the motivation and the scope of this work within the Section Introduction & Background. The state of the art report (see Section State of the Art Report) presents a set of alternative approaches available for enhancing the existing search services. The goals, requirements and specifications that guide the development of the advanced image similarity service are presented in Section Requirements & Specifications, and the details of concrete technical development are described in Section Implementation of advanced image search services and demos. The aggregation of the used dataset and the evaluation of search performance through expert and regular users is discussed in Section Evaluation, while the concluding remarks and the future work directions are expressed in Section Conclusions and Future Work.

[1] http://pro.europeana.eu/files/Europeana_Professional/Projects/Project_list/Europeana_DSI-2/Milestones/ms6.1-advanced-image-discovery-development-plan.pdf

Introduction & Background

Content based search services offer alternative solutions for accessing the information available in large and heterogeneous data repositories. Recent statistics indicate that image search represent one third in google's search requests[2]. This indicators are correlated with the facts that image content is very popular for the web users and content based search is not affected by language barriers. The situation is not different for Europeana portal, where more than 50% of content is represented by image items, and the associated textual descriptions (i.e. metadata fields such as dc:subject and dc:description) are typically available in one of the 37 supported languages. Moreover, many of the image items are associated with poor content descriptions.
The development of image search services was part of several Europeana Network projects. The ASSETS project[3] was the first one to investigate the effectiveness of image search applied to the whole repository. However, the technical limitations of having only 200 pixels thumbnails available was the main impediment for building an effective image search service. Currently 400 pixels thumbnails are available, but also the size of the repository has increased by a factor of 10, which require employment of technologies with higher scalability capabilities.
The image search services developed in Europeana Creative[4] project were build to support the specific functionality within the Design Pilot. These service used a small image dataset (i.e. ~4000 items) which were manually selected according to the visual and technical quality requirements defined by design experts. The resulting CultureCam[5] prototype with its online and interactive installation version received good feedback from the designer community[6].
The development of the Content Reuse Framework[7] was initiated within the same project, and extracts the color histograms that are currently used for filtering content by colors in Europeana Collections.
Within the current project we investigate solutions of advanced image similarity services that are able to provide both a high scalability and stability of the service; and in the same time provide a good accuracy for the similarity search. Additionally, the scalability of the content selection activities is an additional concern, provided that the same process is used for aggregating the evaluation dataset and for scaling up the image index for the consequent versions of the API. The end users are interested in retrieving image content which is rich in information, and they are less interested to retrieve images that represent scans of newspapers or text based posters for example (i.e. even if they may incorporate small design elements). Consequently, a semi-automatic approach for content selection, based on suggestive search queries provided by expert users was employed. Pure manual curation of image content was not an option given the target of aggregating more than 100.000 images for development and extensive evaluation of search performance.

The goal of the Subtask 6.3.4 is to enhance the existing algorithms used for image discovery purpose in Europeana and Europeana related projects, with the aim to enhance their scalability and improving their retrieval accuracy. For improving the scalability and maintenance of the services, we plan to migrate existing implementations to use the mainstream technologies used in Europeana APIs (e.g. Solr-based solutions). In order to enhance the accuracy of search

[2] see https://moz.com/blog/state-of-searcher-behavior-revealed#15: What percent of all searches happen on any major search property in the US?

[3] http://pro.europeana.eu/project/assets
[4] http://pro.europeana.eu/structure/europeana-creative
[5] http://culturecam.eu/
[6] http://pro.europeana.eu/page/design-pilot
[7] http://pro.europeana.eu/files/Europeana_Professional/Projects/Project_list/Europeana_Creative/Deliverables/eCreative_D3.1_KL_v1.0.pdf

algorithms, we plan to integrate state of the art technologies, by evaluating state of the art algorithms developed in the past years, which make use of global and local image features. However, the fine tuning of advanced algorithms requires experimentation with the concrete/targeted Europeana image datasets.

The initial planning of development activities for the advanced image similarity search service was presented within the milestone document MS6.1. The initial plan included the following list of activities, and the slight deviations are discussed in Section Conclusions and Future Work:

- *Content selection for evaluation Dataset*. By using the selection criteria and examples provided in Section Dataset, the image content served by the similarity service is aggregated in a dataset and serialized in a machine readable format. The dataset includes the ids and metadata of Europeana records, and the public URLs made available for downloading the content.
- *Migration of existing image search services*. The image search services developed based on Lucene indexing library are taken in consideration for further development in order to comply with the new technical and functional requirements identified within the current project. In order to ensure a better scalability and management of the image index, the implementation of image search algorithms will be ported to a Solr based solution, possibly by reusing the Solr plugin architecture implemented in the Lire[8] project. This solution will be used as baseline for the experimental evaluation.
- *Implementation of the image search API*. The service must implement a RESTful API using the same technologies as the other Europeana APIs (i.e. based on Spring MVC and Swagger). Additionally, metadata fields used for faceting and filtering purpose need to be integrated into the image index as well.
- *Experimental evaluation*. An experimental evaluation will be carried out to assess the performance of state of the art algorithms when applied on the selected dataset (e.g. including some of the candidate solutions like CDVS, Lire, Amato, Pastec). One of the goals of the comparative evaluation is to evaluate the best performing algorithms for different categories of content (i.e. tailored by subject categories, and technical properties like grayscale vs color images).
- *Advanced image search algorithms*. It is anticipated that different algorithms will be more effective when used in individual search scenarios or when applied on particular content categories (e.g. paintings, fashion artifacts, music instruments, etc.). Consequently, a multi-modal search approach will better serve the user scenarios defined in Section Requirements & Specifications. The novel algorithms need to be based on Solr based solutions in order to be integrated with the baseline solution and support additional filtering/faceting functionality.
- *Integration in Europeana Collections*. The graphical user elements and controls for integrating the functionality of the advanced search in the Europeana Collections need to created for supporting the user stories presented in Section User stories.

State of the Art Report

Europeana Image Search. An image similarity search algorithm, using global image descriptors was developed within the scope of Assets4Europeana project [Amato 2011] and was further developed within the Europeana Creative project, in which the CultureCam frontend was also developed [Gordea 2015]. In this approach, a nearest-neighbour algorithm is used to reduce the search space and implement an efficient indexing solution, that ensures optimal retrieval

[8] http://www.lire-project.net/

performance at run time. This is achieved by selecting a pivot set and computing the distances between each indexed image and the pivots. At runtime, the search results are ordered by their similarity relative to the pivot set. A detailed description of the feature extraction and indexing process is presented in [Amato 2011].

While the global features are not able to catch the particularities of objects available in the image, often, the local features are not appropriate for assessing the overall similarity of the images. As the amount of higher quality images available in Europeana was increasing considerably in the last years and standard thumbnails are available in 400X pixels, it is expected that multimodal search algorithms will improve the search accuracy for the heterogeneous content available in Europeana.

The latest technology development within the research community leans towards the usage of local features for improving the accuracy of image search algorithms. The main scenario used by local feature based search algorithms concentrate on object identification, near duplicate detection and object classification.

In the past years, the MPEG[9] experts group was focusing on creating standardized representations for image descriptions in order to support development of efficient image retrieval systems [Sikora 2001]. While the early standardization work was concentrating on specification of the so called global descriptors, the latest activities focus on local descriptors, which are more appropriate for classification and pattern recognition purposes [Bianco 2015]. The CDVS approach was published as ISO standard[10] in September 2015. A reference software implementation is provided with the standard[11].

There are many different state of the art algorithms available which implement the extraction of local visual features that are used in image recognition and retrieval scenarios. However, in order to support this functionality in client-server environments, supporting various client devices it is important to use standardized solutions. Starting with 2010 the Moving Picture Experts Group (MPEG) worked on creating an ISO standard that specifies the extraction and compression of Compact Descriptors for Visual Search (CDVS) [Duan 2015].

A comparative evaluation of the CDVS approach applied on six different datasets using various variations of the search algorithms is presented in [Bianco 2015]. The search algorithms are based on CDVS specifications but use different algorithms used for detection of interest points. The reported results show very good performance for near-duplicate image search, however, these datasets are much smaller and less heterogeneous than the europeana dataset.

The interest point detection is a core component used in the extraction of local features. SIFT and SURF interest point detectors were developed in the past decades and proved to be very effective for structure and object detection algorithms. However, these are patented solutions and newer detectors like KAZE and ORB were proposed later on. They aim at reducing the computational effort and increase the robustness to noise [Rublee 2011, Alcantarilla 2012]. An open source implementation for various local feature extraction algorithms (i.e. including SIFT, ORB, KAZE) is available in OPENCV library[12]. Pastec[13] is an open source library that implements near duplicate image detection based on indexing of ORB feature descriptors.

[9] http://mpeg.chiariglione.org/

[10] https://www.iso.org/standard/65393.html

[11] http://mpeg.chiariglione.org/standards/mpeg-7/reference-software-conformance-and-usage-guidelines-cdvs/n15371-text-isoiec-cd

[12] http://docs.opencv.org/3.0-beta/modules/features2d/doc/feature_detection_and_description.html

For the implementation of advance image similarity search service, not patented and open source solutions will be taken in consideration.

Lire is implemented as a Java Library that offers an open source implementation of common, mainly global, feature extraction and search algorithms for content based image retrieval[14] [Lux 2008]. The search engine is implemented on the top of Lucene library which is well known as the main open source text retrieval library. In 2015 the beta version of the Lire Solr plugin was released, which exploits the engineering work invested by the Solr community. Concretely, this plugin has the goal to offer a better scalability and configurability for Lucene based search indexes by making them cloud compliant. Lire offers support for extraction of various visual image descriptors and it also offers an implementation for various similarity measures, which can be used in combination with specific features. However, the retrieval accuracy of image retrieval algorithms is highly dependent on the characteristics of the given datasets and by appropriate combination of feature descriptors and weighting schemes. A comparative evaluation of various search algorithms applied on 4 different datasets is presented in [Lux et al. 2016]. However, these experiments present the performance of available feature descriptors, but they do not address the effect of combining different features in an advanced search algorithm.

In the context of advanced image retrieval service, the feature extractors and the Solr plugin are the most relevant functionality offered by Lire. A Solr based implementation of the search algorithms will ensure the required scalability of the search service. Moreover, this will provide support for integrating similarity search with filtering and faceting functionality. An open question is related to possibility of porting the different indexing solutions to a Solr/Lucene based implementation.

Requirements & Specifications

The advanced image similarity search service is meant to be implemented as a core service that will serve different applications and user scenarios. Consequently, the development of the service will be guided by the particular functional needs expressed in terms of user scenarios and information needs expressed in terms of content selection criteria (see Dataset).

Within the Europeana CultureCam[15] product briefing a product vision is presented. This has the goal of supporting the acquisition of concrete functional requirements for the image search services. This document fosters the long term ambitions on integration of image search functionality within the Europeana Collections, which go beyond the timeframe of the current project.

User expectations

The user research carried out by Europeana indicates that there is a small[16] audience of designers and artists who use Europeana content. They find images from Europeana to inspire their work or to reuse them when creating own works. A small number of this type of users were

[13] http://pastec.io/

[14] http://www.semanticmetadata.net/wiki/

[15] https://docs.google.com/document/d/1hRHHLEhD4jSTPvBZo0yiFZntJVBy7GeZGHiRjCgDfPg/edit#

[16] About 10%. Numerically small but it’s an audience segment we want to grow and it’s also an audience segment that does something we’re very committed to: *use our shared heritage to create new works that may become our shared heritage in the future!*

interviewed in the past to discover their interests and expectations (e.g. see blog post by Kumiko Sakaki).

Differently from professionals, the public users are interested in alternative ways of navigating through large repositories like Europeana. They are typically interested in more exploratory ways of browsing the collections; a way that is more image oriented and tailored their needs. A navigation that is better suited for those who don't think in keywords and thus find an empty search box a poor starting point[17]. They have an idea of what they're looking for, but they don't know exactly what it is. We want to help them finding what they didn't knew, at the time they start the exploration of Europeana repository.

The personas Marie and Linda were created in the briefing document to facilitate the analysis of these two categories of users.

Consequently, a set of three user stories was introduced in MS6.1 document. The first two of them concentrate on the classic scenarios of content based retrieval, namely search similar images to a query (image) available in the dataset (i.e. story: *Find similar to existing Europeana images*); or search similar images to one provided by the user (i.e. story: *Find similar to user provided images*). The third scenario concentrates on refining the search results by exploiting (text based) faceting and filtering functionality (i.e. story: *Filtering search results*). More details on these scenarios can be found in the Europeana CultureCam product briefing.

Note that while these user stories are phrased from the perspective of a user of Europeana Collections, they indirectly set the requirements on the Image similarity API and backend. For every user story, the user might as well be a developer that integrates this service in third party applications.

[17] What George Oates, a proponent of exploratory browse oriented, library and museum collections UIs call: "The tyranny of the empty search box".

Implementation of advanced image search services and demos

The main goal of the advanced image search service is to build an API, which is publicly available and ready to be integrated in Europeana Collections and third party applications. The general development process follows the Europeana guidelines for API development and quality assurance, using the provided technical infrastructure for continuous integration and testing purposes. For evaluating the search performance of the service, a demonstrator and an evaluation graphical user interface were developed. The details on the activities carried out to accomplish these targets are presented in the following subsections.

Development Process

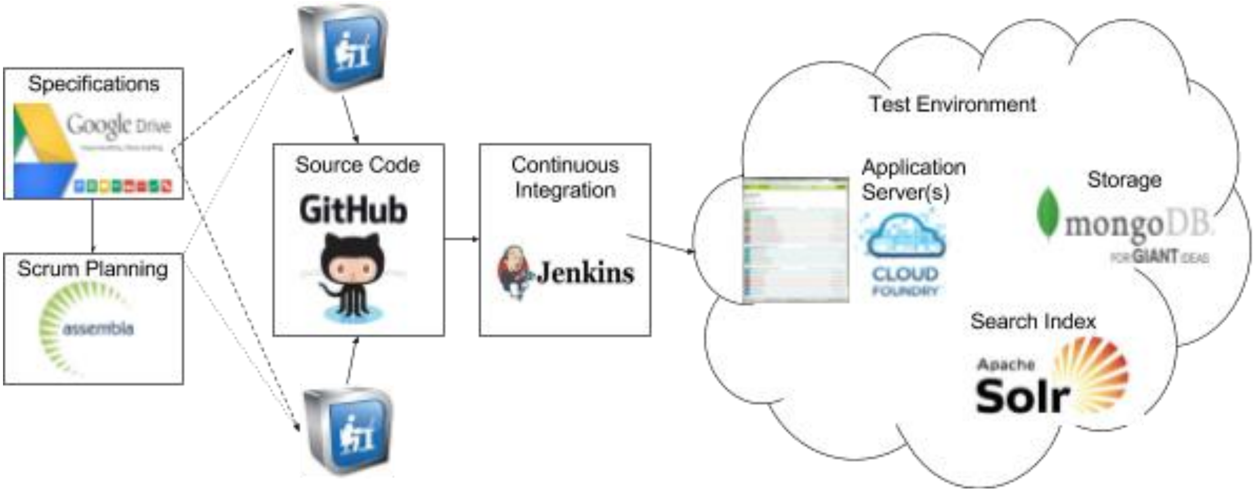


Figure 1: Development process and continuous integration infrastructure

The process and the infrastructure used for the development of Advance Image Similarity Services follows the common approach used in the other Europeana APIs (i.e. search, annotations, entity, etc.).

AIT and Europeana are collaborating for creating and reviewing the software requirements and specifications of the API. When using a distributed team, an agile development process and appropriate set tools needs to be used for managing development and quality assurance activities. Figure 1 is sketching the overall process and technical infrastructure used to support it, depicting the following activity workflow:

- The requirements and specifications for the functionality implemented by the API are collected using Google Drive/Docs, which offers support for real-time editing, annotating and commenting text documents. The first version of the specifications was creating by starting from the functionality offered in the LireSolr demo application. The developers are reviewing the specifications and the critical comments are solved by reaching a consensus. In the end of this first step, the concrete functionality is planned for development.

- The planning of all API related activities is managed using the Assembla system, which is an online tool used for organizing tasks according to Scrum and Sprint based development guidelines (i.e. including specification, development, testing, etc.).
- The generated source code is open source, open licenced (under the EUPL, Version 1.1) and publicly available in Github repositories.
- Jenkins is a continuous integration tool, used for automatic building and deployment of software artefacts.
- The Test Environment uses a cloud-based infrastructure to deploy the API and make it available for testing.
- The search index is based on scalable SolrCloud technology, to which the LireSolr plugin was deployed. The default Solr Admin console is used for managing, monitoring and testing the image index.

Specifications

Inspired by the previous experiences with image search services employed within the previous Europeana Network projects, especially from the Europeana Creative, a document for the vision and longer term ambitions was created to guide the specifications and development processes. The Europeana Culture Cam product briefing is a working document available online[18]. This document goes beyond the scope of EDSI2 Project, but it offers a good overview of the context in which the API may to be used. Therefore, this document is used for deriving the first requirements for the Similarity API[19] specifications. The later document describes the main functionality to be implemented at API level. There are three main search scenarios identified to be covered by the similarity api:

- search by europeana record id, which retrieves images similar with the thumbnail of the provided record
- search by web url, which retrieves images similar to an image available on the web
- search by user provided image, which retrieves images similar to the image uploaded by the user from his/her local machine

The API specifications include, also, the non functional requirements, which are used to align the design and the responses of the Similarity API with the other Europeana APIs. Consequently, this has the goal to enhance the quality of the service and improve the experience of third party developers that make use of these tools. Particularly, the input and the output used in different APIs must be consistent in format and terminology. Additionally, the best practices and technologies used for web development provide guidelines for the headers used in web requests/responses and effective exception handling mechanisms. These aspects are also covered within the specifications document.

[18] https://docs.google.com/document/d/1hRHHLEhD4jSTPvBZo0yiFZntJVBy7GeZGHiRjCgDfPg/

[19] https://docs.google.com/document/d/1z5SGBFw7LT3XqZXNq9zUyEQjjemzztRgr9kexcK9HbQ/

Application Architecture

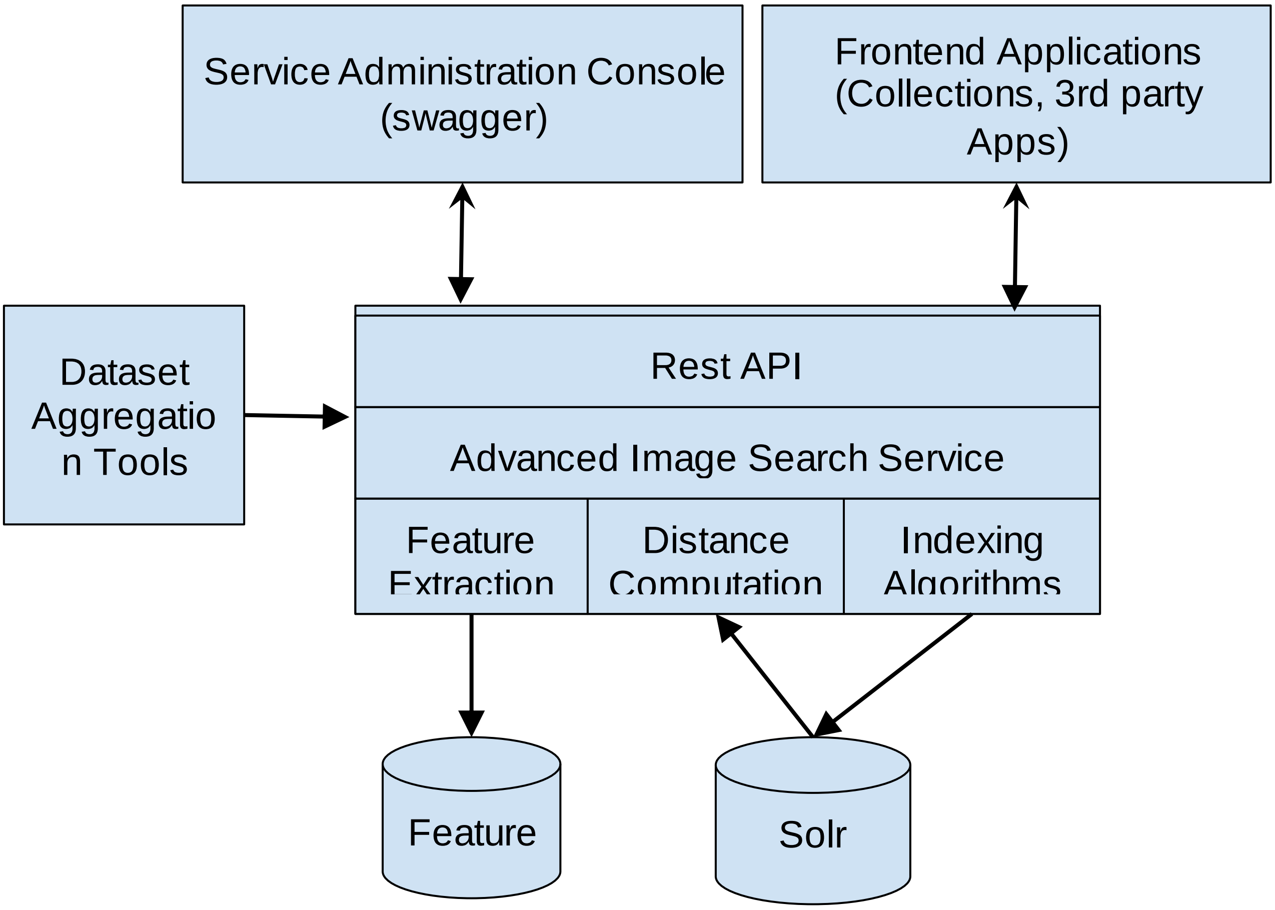

Service Administration Console (swagger)
Frontend Applications (Collections, 3rd party Apps)
Dataset Aggregation Tools
Rest API
Advanced Image Search Service
Feature Extraction
Distance Computation
Indexing Algorithms
Feature
Solr

Figure 2: Architecture overview

The application architecture that shows the main building blocks of the Advanced Image Search Service is presented in Figure 2. It includes also the dataflows exchanged with the Administration Console, dataset aggregation tools and the available Frontend applications.

The advanced image search service is to be integrated within the Europeana APIs, and consequently it was design to use the same technology stack (i.e. Spring and Solr based technologies). The runtime environment is empowered with a REST interface which is built by using Springfox/Swagger libraries. The business logic layer is represented by the Advanced Image Search Service, which follows the Facade[20] design pattern and offers a clear and simple Java interface for accessing its functionality. The concrete implementation of this interface is built on the top of existing solutions available in image search libraries. The lower level functionalities of the service include the extraction of image features (i.e. Feature Extraction), computation of distances and similarities (i.e. Distance Computation) and the transformations required for storing them in a regular search index (i.e. Indexing Algorithms).

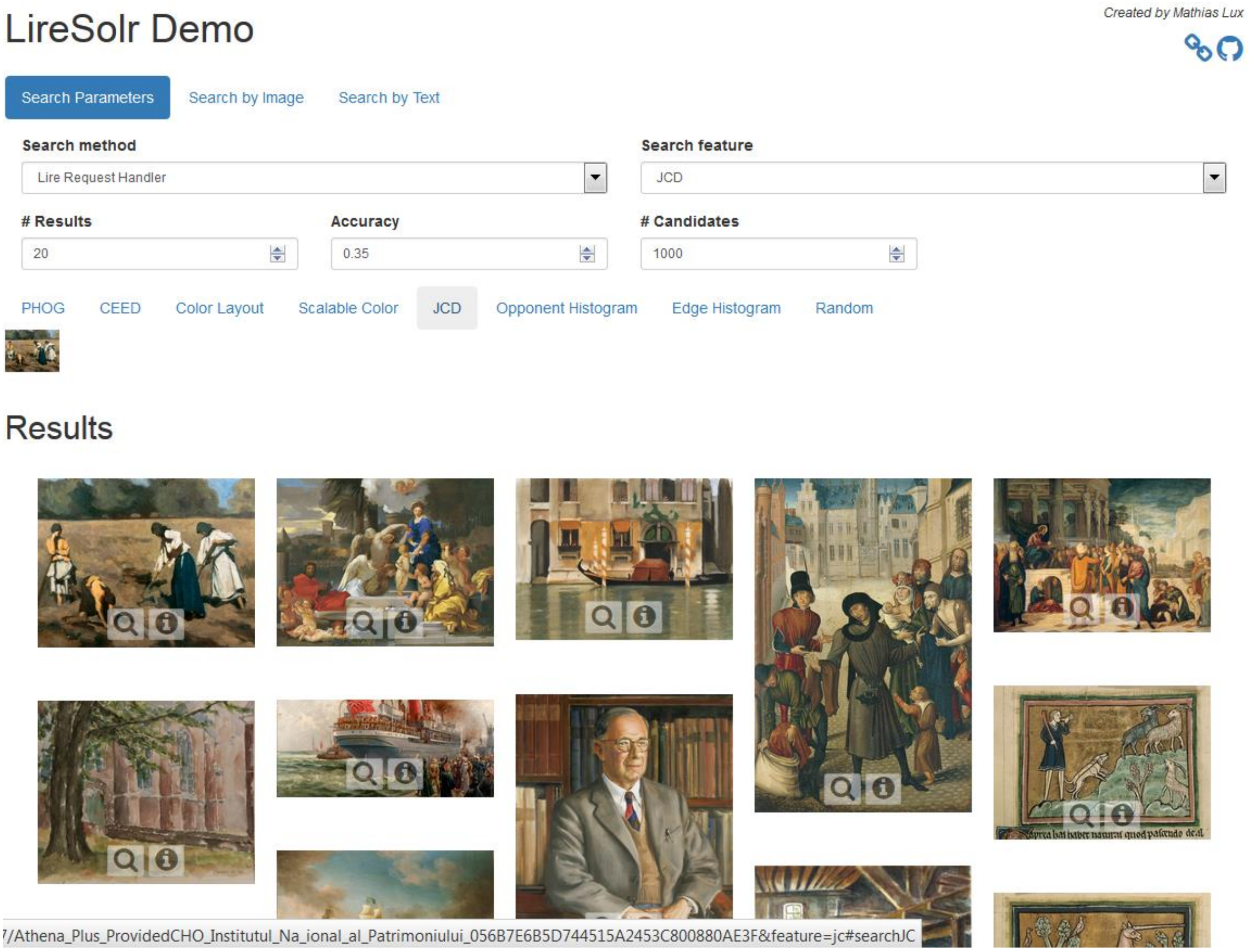
In order to ensure service scalability and runtime performance, the Solr system is used to store image distances and to compute the result sets for similarity search requests. The API is meant to be integrated in Frontend Applications, including Europeana Collection, Demo apps (e.g. evaluation frontend) or third party applications (e.g. future versions of CultureCam.eu). The

[20] https://en.wikipedia.org/wiki/Facade_pattern

Swagger console offers a handy GUI interface for testing and debugging the functionality of the API. The console is a valuable support for developers implementing client applications.
The data stored in the Solr index includes the Thumbnail urls and selected metadata from Europeana repository. This information is acquired by using Europeana search API and the Dataset Aggregation Tools which export the required data in machine readable text files.

Image Similarity Demo and Evaluation Application

The preliminary evaluation of the alternative image search algorithms provided out-of-the-box in LireSolr is performed using the demo graphical user interface available in LireSolr. This was enhanced by adding more features into the index, to use the europeana record ids, to link them with the items in Europeana portal and to display basic metadata attached to the image items.



LireSolr Demo

Created by Mathias Lux

Search method: Lire Request Handler

Search feature: JCD

Results: 20

Accuracy: 0.35

Candidates: 1000

Results

7/Athena_Plus_ProvidedCHO_Institutul_Na_ional_al_Patrimoniului_056B7E6B5D744515A2453C800880AE3F&feature=jc#searchJC

Figure 3: Extended Demo Application

In order to facilitate the user evaluation, the demo interface was extended with additional functionality which allows the end users to perform the same search using alternative algorithms (see Figure 3) .

For the test and demo environment the underlying Solr environment was updated to version 6.5.1[21], owing to issues related to data import encountered with earlier versions of the application. As a result of this upgrade, importing image similarity data is now stable, reliable, and efficient.

[21] http://lucene.apache.org/solr/6_5_1/changes/Changes.html

Implementation of Similarity API

Within the initial activity plan it was envisioned to migrate the search service implemented within the Europeana Creative project to a Solr based solution. This service is based on the approximate indexing based on metric spaces proposed by Giuseppe Amato [Amato2002] and was successfully applied on Europeana content [Amato 2011].

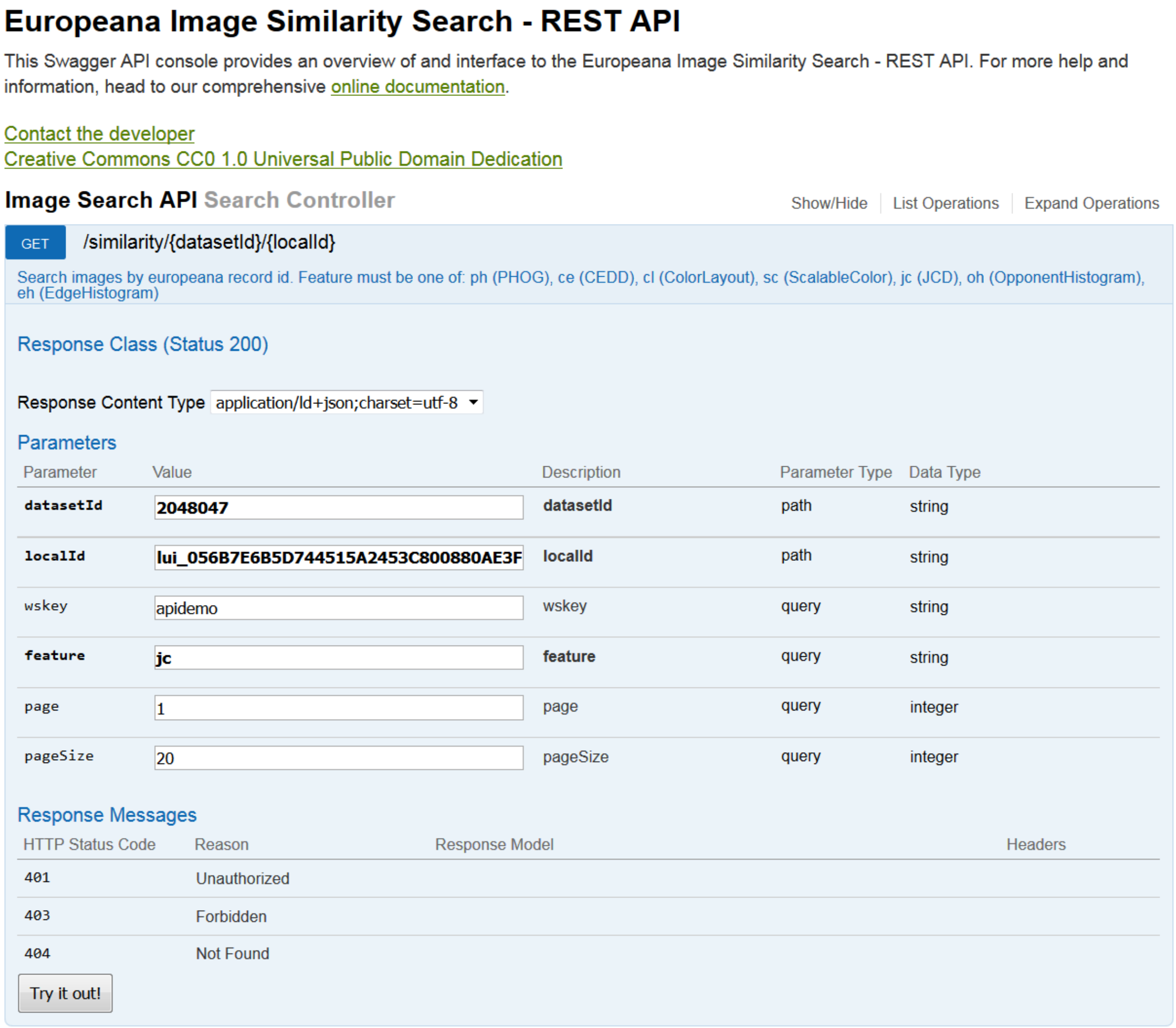
The technical evaluation of the LireSolr[22] plugin, however, indicated that equivalent functionality for approximate indexing and similarity search is now available in this library[23]. By taking in account the following technical considerations, we decided that LireSolr is the more appropriate to be used as basis for the development of the Similarity API:

- The existing image similarity search solution from Europeana Creative[24] uses an older version of Lire, which is exclusively used for feature extraction purposes. It also uses an outdated version of Lucene, namely 4.10.
- LireSolr is based on a newer version of Lire which is able to extract and index a larger palette of visual features[25].
- LireSolr is based on newer Solr Lucene version which includes significant performance and scalability improvements.
- LireSolr is implemented as a solr plugin, which has the advantage of administering the search index using the standard administration console shipped with Solr, and is also more effective when integrating text based faceting and filtering functionality in the API.

However, the LireSolr library does not provide built-in functionality for integrating textual metadata within the search index. A fork of LireSolr was created on Github for implementing the needed functionality[26]. Concretely, the indexer was updated so that the europeana record id is used as identifier for the thumbnails and metadata provided as comma separated value files is included in the XML serialization of Solr input documents.

The API is empowered with a Swagger based console, which is used for testing, to provide a playground and documentation for third party developers. The image similarity API is deployed in the Europeana test environment[27] offering a public Endpoint for accessing its functionality (i.e including the Swagger console). The documentation of the API will be available in the Europeana Labs once the migration of labs pages to the new CMS will be completed[28].
In Figure 4 and Figure 5 present the screenshots of the Swagger console illustrating the search by europeana record id functionality.
The same similarity search request is performed as the one that is shown in a user friendly representation in Figure 3 (i.e. through the evaluation graphical interface). This API method takes the record ID as parameter in the URL, which is split in the console into two variables, the europeana *datasetId* and the *localId* of the europeana object. The *wskey* parameter represents the API key used to control the access of client applications to the API functionality, while the *feature* parameter represents the short name of the image feature used by the search algorithm. The *page* and *pageSize* parameters are used for navigating through the search results by using a standard pagination functionality.

[22] http://www.semanticmetadata.net/2017/01/02/gradle-as-build-system-for-lire-liresolr/
[23] http://www.semanticmetadata.net/wiki/#how-does-lire-actually-work
[24] https://github.com/europeana/Europeana-Creative/tree/master/image-similarity
[25] http://www.semanticmetadata.net/wiki/
[26] https://github.com/gsergiu/liresolr
[27] http://test-image-similarity.eanadev.org/
[28] to be available under: http://cope.eanadev.org/page/image-similarity-api



Europeana Image Similarity Search - REST API

This Swagger API console provides an overview of and interface to the Europeana Image Similarity Search - REST API. For more help and information, head to our comprehensive online documentation.

Contact the developer
Creative Commons CC0 1.0 Universal Public Domain Dedication

Image Search API Search Controller

/similarity/{datasetId}/{localId}

Search images by europeana record id. Feature must be one of: ph (PHOG), ce (CEDD), cl (ColorLayout), sc (ScalableColor), jc (JCD), oh (OpponentHistogram), eh (EdgeHistogram)

Response Class (Status 200)

Response Content Type: application/ld+json;charset=utf-8

Parameters

Parameter	Value	Description	Parameter Type	Data Type
datasetId	**2048047**	**datasetId**	path	string
localId	**lui_056B7E6B5D744515A2453C800880AE3F**	**localId**	path	string
wskey	apidemo	wskey	query	string
feature	**jc**	**feature**	query	string
page	1	page	query	integer
pageSize	20	pageSize	query	integer

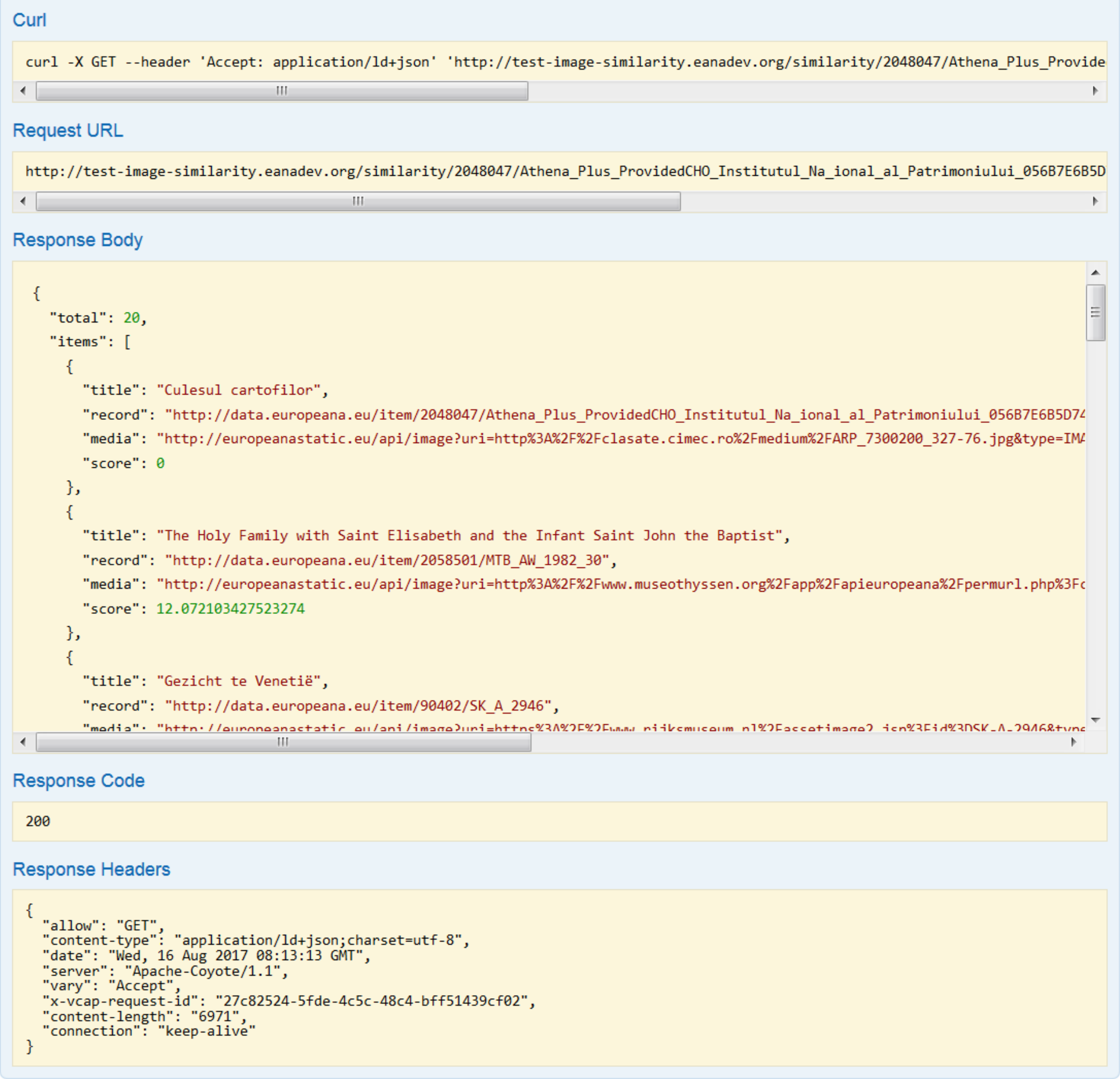
Response Messages

HTTP Status Code	Reason	Response Model	Headers
401	Unauthorized		
403	Forbidden		
404	Not Found		

Figure 4: Swagger Console

The API response for the given search has a developer friendly representation in the Swagger console (see Figure 5). The equivalent *curl* command and the *Request URL* represent two alternative possibilities for remote invocation of the API method. The first one is based on a command line tool used to build web requests[29], while the second represent the address of the web resource that can be used in standard web browsers. The Response Body includes the payload of the HTTP response, displaying the API response serialized in json format. *total* represents the number of search results included in the current page, while the *items* themselves are serialized by including their *title*, the web reference to the europeana *record,* the web location of the *media* file representing the thumbnail of the item, and the *score* representing the computed distance between the query image and the retrieved one. Note that in the search by record id method, the first retrieved result is the same as the query image and the computed distance is 0 by definition.
The HTTP *response code* (i.e. also known HTTP status code) and the *response headers* are shown in the console as well, being used by client applications for debugging and exception handling purposes.

[29] https://en.wikipedia.org/wiki/CURL



Curl

```
curl -X GET --header 'Accept: application/ld+json' 'http://test-image-similarity.eanadev.org/similarity/2048047/Athena_Plus_Provide
```

Request URL

```
http://test-image-similarity.eanadev.org/similarity/2048047/Athena_Plus_ProvidedCHO_Institutul_Na_ional_al_Patrimoniului_056B7E6B5D
```

Response Body

```
{
  "total": 20,
  "items": [
    {
      "title": "Culesul cartofilor",
      "record": "http://data.europeana.eu/item/2048047/Athena_Plus_ProvidedCHO_Institutul_Na_ional_al_Patrimoniului_056B7E6B5D74
      "media": "http://europeanastatic.eu/api/image?uri=http%3A%2F%2Fclasate.cimec.ro%2Fmedium%2FARP_7300200_327-76.jpg&type=IMA
      "score": 0
    },
    {
      "title": "The Holy Family with Saint Elisabeth and the Infant Saint John the Baptist",
      "record": "http://data.europeana.eu/item/2058501/MTB_AW_1982_30",
      "media": "http://europeanastatic.eu/api/image?uri=http%3A%2F%2Fwww.museothyssen.org%2Fapp%2Fapieuropeana%2Fpermurl.php%3Fc
      "score": 12.072103427523274
    },
    {
      "title": "Gezicht te Venetië",
      "record": "http://data.europeana.eu/item/90402/SK_A_2946",
      "media": "http://europeanastatic.eu/api/image?uri=https%3A%2F%2Fwww.rijksmuseum.nl%2Fassetimage2.jsp%3Fid%3DSK-A-2946&type
```

Response Code

```
200
```

Response Headers

```
{
  "allow": "GET",
  "content-type": "application/ld+json;charset=utf-8",
  "date": "Wed, 16 Aug 2017 08:13:13 GMT",
  "server": "Apache-Coyote/1.1",
  "vary": "Accept",
  "x-vcap-request-id": "27c82524-5fde-4c5c-48c4-bff51439cf02",
  "content-length": "6971",
  "connection": "keep-alive"
}
```

Figure 5: Swagger Console

The source code of the image similarity API is publicly available in Europeana Github repository[30]. The generic implementation of technical and infrastructural requirements is shared between the Annotation API, Entity API and Image Similarity API, therefore it is made available as support libraries having the source code[31] publicly available in Github.

[30] https://github.com/europeana/api-image-similarity

[31] https://github.com/europeana/api-commons

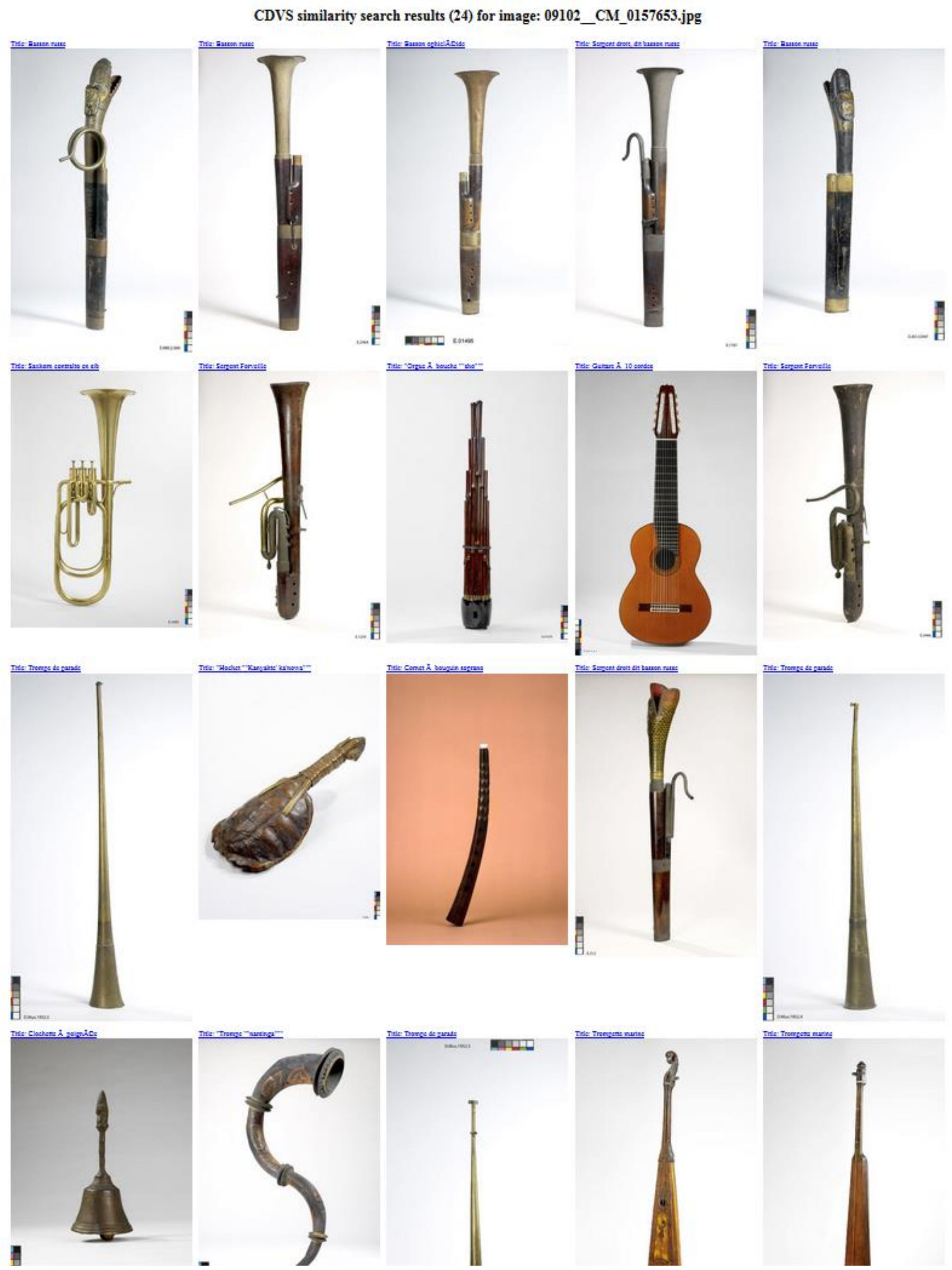
Image search using local features

The CDVS matching algorithm takes in account the local features extracted from the images and their geometry, and it is primarily used to identify if two images may represent the same objects or not. However, the number of matching features and a numeric distance are computed, which can be used for ranking items in a similarity search scenario.
A technical and functional evaluation regarding the feasibility of including the CDVS based search in the image similarity API was performed within the scope of this task. We were particularly interested in the scalability of the existing software, possibility to integrate it in a LireSolr and the search scenarios in which this approach provides a high accuracy on the given dataset. The results of this evaluation are summarized in the followings:

- The evaluation available in the CDVS conformance testing software is used for the CDVS provided image set, which contains c.a. 500 images[32]. The computation of the image similarities is computation intensive and is results are not stored in an index. Therefore, additional engineering efforts are required to ensure the scalability level required by the europeana requirements.
- Offline computation of similarities/distances using the CDVS approach and indexing using LireSolr would be possible, however this solution is not supporting the search by external images scenarios. A performing implementation that supports this functionality requires building of an alternative indexing and retrieval solution, which is tailored to the capabilities of Solr/Lucene indexers.
- The retrieval functionality, called matching in CDVS terms, is build to support near duplicate detection scenario. Therefore, the provided scoring function is optimized for detecting exact matching objects and not on ranking images with loose similarity.
- The CDVS search is based on luminance information available in the images, therefore the extracted features belong to the shape based category. As result, a CDVS search often includes grayscale and colorful images in the search results. This is sometimes not what the users expect as outcome for similarity search. However, the advanced algorithms combining CDVS with global, color based, features may provide significant improvements of search accuracy.

Given the fact that the algorithm is based on shape information, the resulting search algorithm is primarily effective to search for images that contain objects, like music instruments, jewelries, shoes, etc. In Figure 6 we present the search results when using a music instrument as query, namely a russian trumpet (i.e. the top left image in the result set represents the search query). For this particular search query, all results in the Top 24 (i.e. default page size of europeana results) list are populated by music instruments, most of them being trumpets and the other's sharing a certain level of form similarities (see Figure 6).

[32]



CDVS similarity search results (24) for image: 09102__CM_0157653.jpg

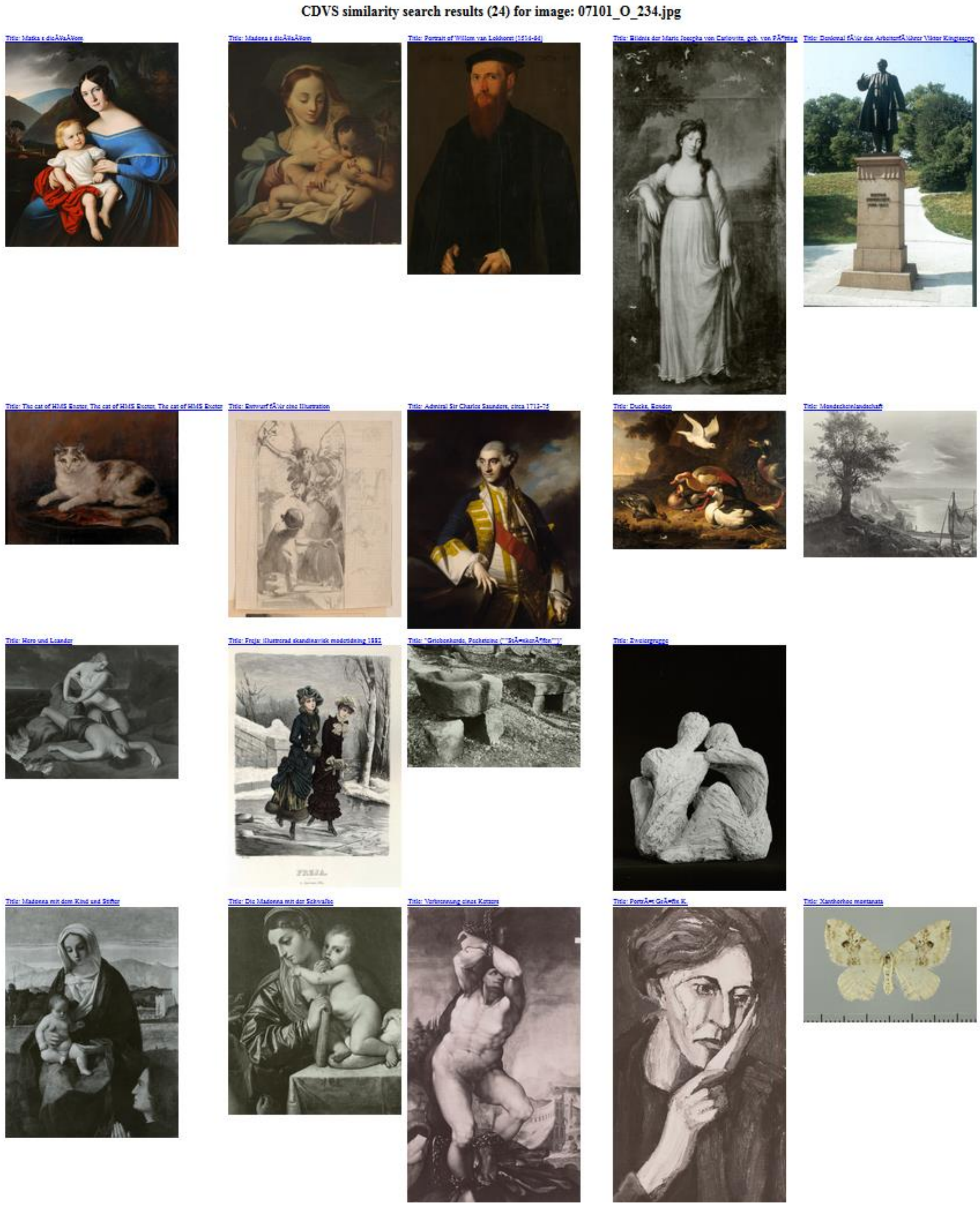
Title: Basson russe
Title: Basson russe
Title: Basson ophiclÃ©ide
Title: Serpent droit, dit basson russe
Title: Basson russe
E.01495
Title: Saxhorn contralto en mib
Title: Serpent Forveille
Title: "Orgue Ã bouche ""sho"""
Title: Guitare Ã 10 cordes
Title: Serpent Forveille
Title: Trompe de parade
Title: "Hochet ""Kanyahte' ka'nowa'"""
Title: Cornet Ã bouquin soprano
Title: Serpent droit dit basson russe
Title: Trompe de parade
Title: Clochette Ã poignÃ©e
Title: "Trompe ""narsinga"""
Title: Trompe de parade
Title: Trompette marine
Title: Trompette marine

Figure 6: CVDA Search Result - Russian Trumpet as query image

Another search scenario in which local features may provide better results that the global descriptors is search for common topics/motifs. For example the Virgin with the Child is a common motif, that was present in many forms in many famous paintings, some of them being created in first centuries of our era[33]. Figure 7 presents the search results using a slovakian painting representing the Virgin with the Child. Within the Top 24 results, the first two items represent the same motif, being colorful images and having a similar painting style. On the 20th and 21st position we find grayscale images representing again the Virgin with the Child, even if they are more different with regard to the painting style. In comparison, the standard algorithms

[33] https://en.wikipedia.org/wiki/Madonna_(art)

included in the similarity API are not able to retrieve images with the same motif, except the color layout based algorithm which is able to find one additional item[34].



CDVS similarity search results (24) for image: 07101_O_234.jpg

Title: Matka s dicÂ¼aÂ¼em

Title: Madona s dicÂ¼aÂ¼em

Title: Portrait of Willem van Lokhorst (1514-84)

Title: Bildnis der Marie Josepha von Carlowitz, geb. von PÃ¶ting

Title: Denkmal fÃ¼r den ArbeiterfÃ¼hrer Viktor Kingissepp

Title: The cat of HMS Exeter, The cat of HMS Exeter, The cat of HMS Exeter

Title: Entwurf fÃ¼r eine Illustration

Title: Admiral Sir Charles Saunders, circa 1713-75

Title: Ducks, Bondon

Title: Mondscheinlandschaft

Title: Hero und Leander

Title: Freja: illustrerad skandinavisk modetidning 1882

Title: "Griebenkunde, Pochsteine (""StÃ¤nkerÃ¶fen"")"

Title: Zweiergruppe

Title: Madonna mit dem Kind und Stifter

Title: Die Madonna mit der Schwalbe

Title: Verbrennung eines Ketzers

Title: PortrÃ¤t GrÃ¤fin K.

Title: Xanthorhoe montanata

[34] http://image-similarity.ait.ac.at/solr/lire_eval.html?id=/07101/O_234&feature=cl

Figure 7: CVDA Search Result - Madonna With Child as query image

Europeana CultureCam Product Idea

The original objective of CultureCam was to make it fun, addictive and inspiring to browse a selection of high-quality downloadable clearly rights labelled images in Europeana Collections, based on images you already like or find intriguing. This was inspired from the existing CultureCam.eu prototype and extending to reach a different audience and to use a larger dataset of images available for search. The new created product briefing document (see Section Specifications) lists the possible user scenarios, targeted audience and proposes a set of wireframes that showcase a possible interaction with the graphical user interface (see Figure 8).

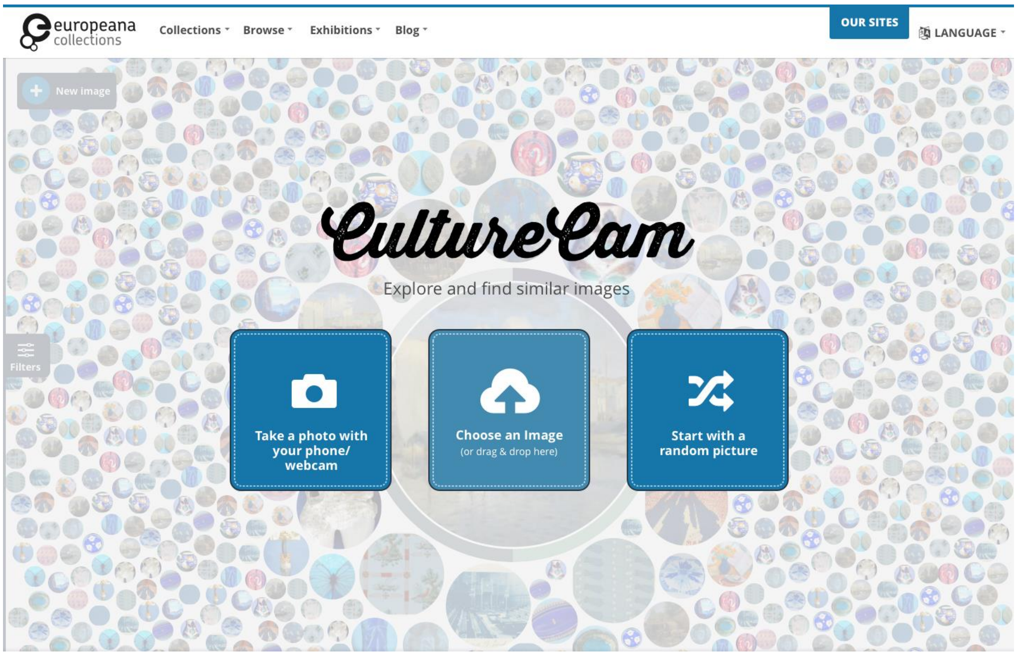

europeana collections
Collections
Browse
Exhibitions
Blog
OUR SITES
LANGUAGE
New image
Filters
CultureCam
Explore and find similar images
Take a photo with your phone/ webcam
Choose an Image (or drag & drop here)
Start with a random picture

Figure 8: Wireframe for CultureCam within the product briefing document

The main goal of the briefing document is to showcase browsing and exploration opportunities opened by the image similarity API. The proposal fed the requirements for image discovery, but later it was decided to link the concrete GUI implementation into the new item page designs. The functional prototypes are expected to be performed within the next iterations of development in Europeana Collections.

Evaluation

The similarity search algorithms typically need to make a tradeoff between the server response time and the precision of retrieval. A higher accuracy of the search is obtained when combining different features, using both global and local descriptors. The server response time is mainly depending on the complexity of the similarity measures, the complexity of feature descriptors and the method used for indexing the extracted features. Therefore one must choose a method carefully, and the evaluation of competing solutions is one of the core activities for building advanced image search algorithms.

The evaluation of alternative out-of-the-box algorithms provided with LireSolr was divided in two parts. The preliminary evaluation is based on expert opinion which has the goal of evaluating the strengths and the weaknesses of the individual search algorithms with respect to the given image dataset. This first step of the evaluation was performed by internal stakeholders and was not strongly formalized, being relatively informal. However, it collected valuable feedback on the performance of available search algorithms and was later used a basis for defining the methodology applied in the second step of the evaluation, which is run with external users.

The aggregation of the image dataset used and the procedures used for evaluation purposes are presented in the following subsections.

Dataset

The first step within the implementation of advanced image search services is the selection of the an appropriate dataset. The quality of the search results is directly dependent by the technical quality of the images. Typically, the image processing and retrieval services recommend an optimal size for the processed images, which lays around 500 pixels on the smallest dimension (i.e. height or width). The algorithms are able to work with smaller images, however for optimal retrieval is recommended that the size of used images does not deviate much from the recommended value. When using high resolution images the processing time increases considerable without providing a significant increase of the search accuracy. Consequently, from practical reasons, we chose to use the 400px thumbnails available in Europeana

The color information, and the quality of content has an important impact on the performance of most image retrieval algorithms, and on the similarity as perceived by the end users. With respect to this aspect, the content available in the Europeana Thematic Collections (i.e. Art History and Fashion) was identified to be best suited to be used within this service.

The guiding principles for the selection of image dataset include:

1. Image content types (e.g. subject categorizations) must be aligned with one or more themes and thematic collections (i.e. Art, Fashion, Music are top priority, other themes may be added later)
2. Image content types must be specific enough and provide meaningful information to the end user (e.g. Clothing, not Fashion)
3. Image content types should not be too narrowly defined as that would be very difficult to scale (e.g. Music instruments, not Flutes)

The complete list of search queries used for selection of image content for the similarity API was collected in Annex 2.

Note that all suggested links/selections include some level of noise, in the sense of false positives associated to an image content type will be included in the results. This cannot be avoided when content selection is based on free text search and no consistent categorization schema is used within the metadata (e.g. in subject field for example) with respect to the defined content types. This noise can be eliminated only through a manual curation process, which is a time consuming approach and is not scalable for the targeted size of the dataset. A mixed approach may be used in the future, in which the initial selections represent hand picked training sets for automatic classification using machine learning algorithms.

Expert Evaluation

Given the heterogeneity of the dataset, two internal experts were evaluating the search performance using different types of query images, quantifying the overall similarity and serendipity level of the search result. While the retrieval of similar images is the main purpose of this service, the serendipity level indicates the amount of unexpected, but relevant information included in the search results [Hill 2016]. In the case of search results that include exclusively very similar images the users are tempted to classify these results as boring, which directly impacts user's motivation to use the given system. Even if there were no explicit quality criteria defined for assessing the search performance, the expectations of these users in term of good results were quite similar, given the past experiences gained through the evaluation of CultureCam prototype developed in Europeana Creative project[35].

By using the demo graphical user interface, the expert users were navigating on random bases through the image dataset. They selected different types of images (i.e. images with various color distributions, represented different motifs) for evaluating the performance of individual search algorithms for the given query.

The first expert was using a comprehensive spreadsheet[36] document to collect different insights of individual search examples by describing the query images and evaluating the quality of the search results in terms of overall search accuracy, serendipity and ability to detect near duplicates. Figure 9 presents the overall assessment computed for the 13 search queries. As expected the color based features (i.e. JCD, ColorLayout, CEDD) are the ones providing the best search results, however, for colorless images, the Edge Histogram was able to provide best results.

[35] http://pro.europeana.eu/files/Europeana_Professional/Projects/Project_list/Europeana_Creative/Deliverables/eCreative_D6.3_platoniq_v1.0.pdf

[36] https://docs.google.com/spreadsheets/d/1i5sEYUbZ8T5jcvzbJzUpHJ3D7Eyoxo_5fpWIB9YtLZg/edit#gid=0

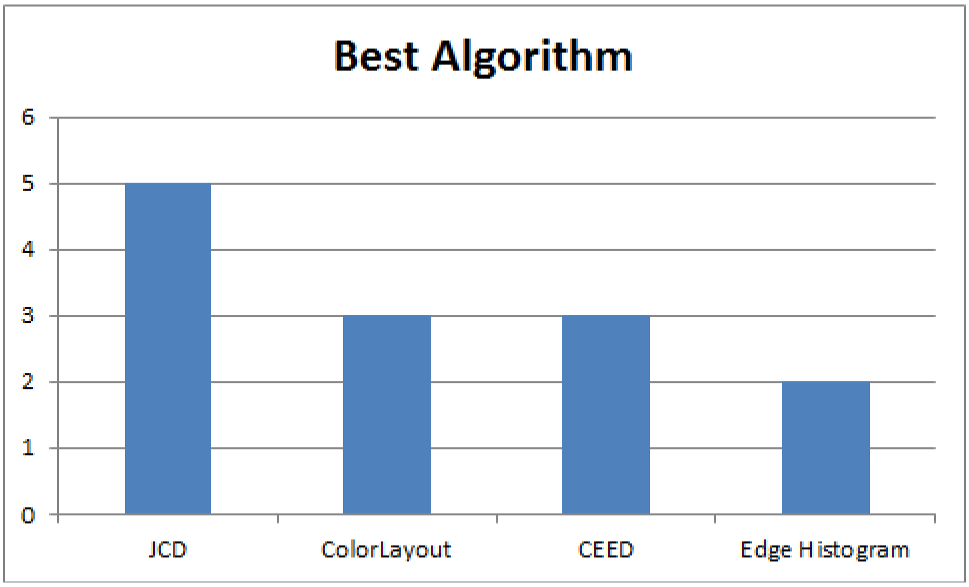


Figure 9: Counting the cases when an algorithm was indicated as being the best one

The distribution of best algorithms over the content categories is presented in Figure 10, indicating that specific algorithms are appropriate for searching a certain category of content. In particular ColorLayout and CEED are appropriate to search small objects like shoes, jewelries or decoration objects and JCD is appropriate for images rich in colour and form information, like paintings or posters. Edge Histogram can be used successfully to retrieve colorless images such as sketches or music instruments. Color layout is also effective in retrieving fotos representing or portraits of people.

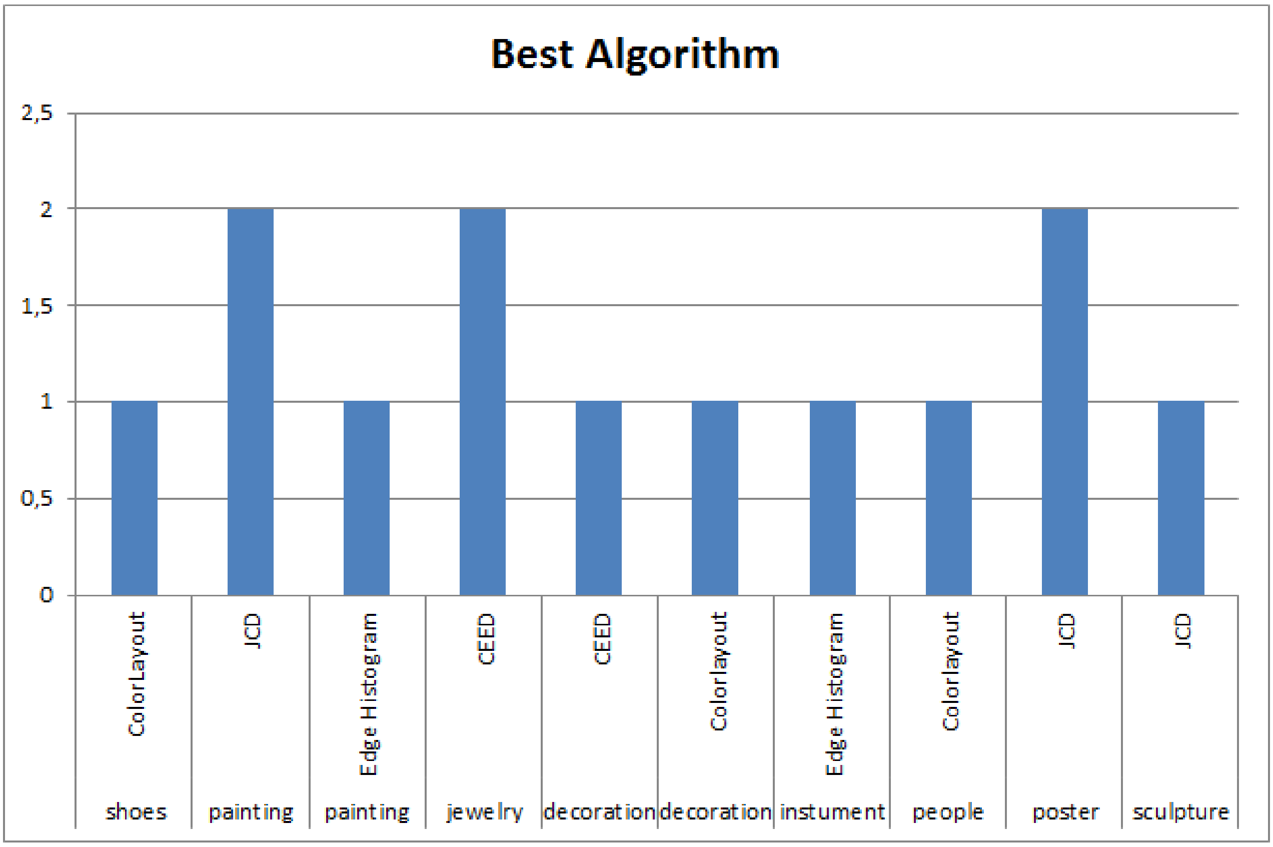
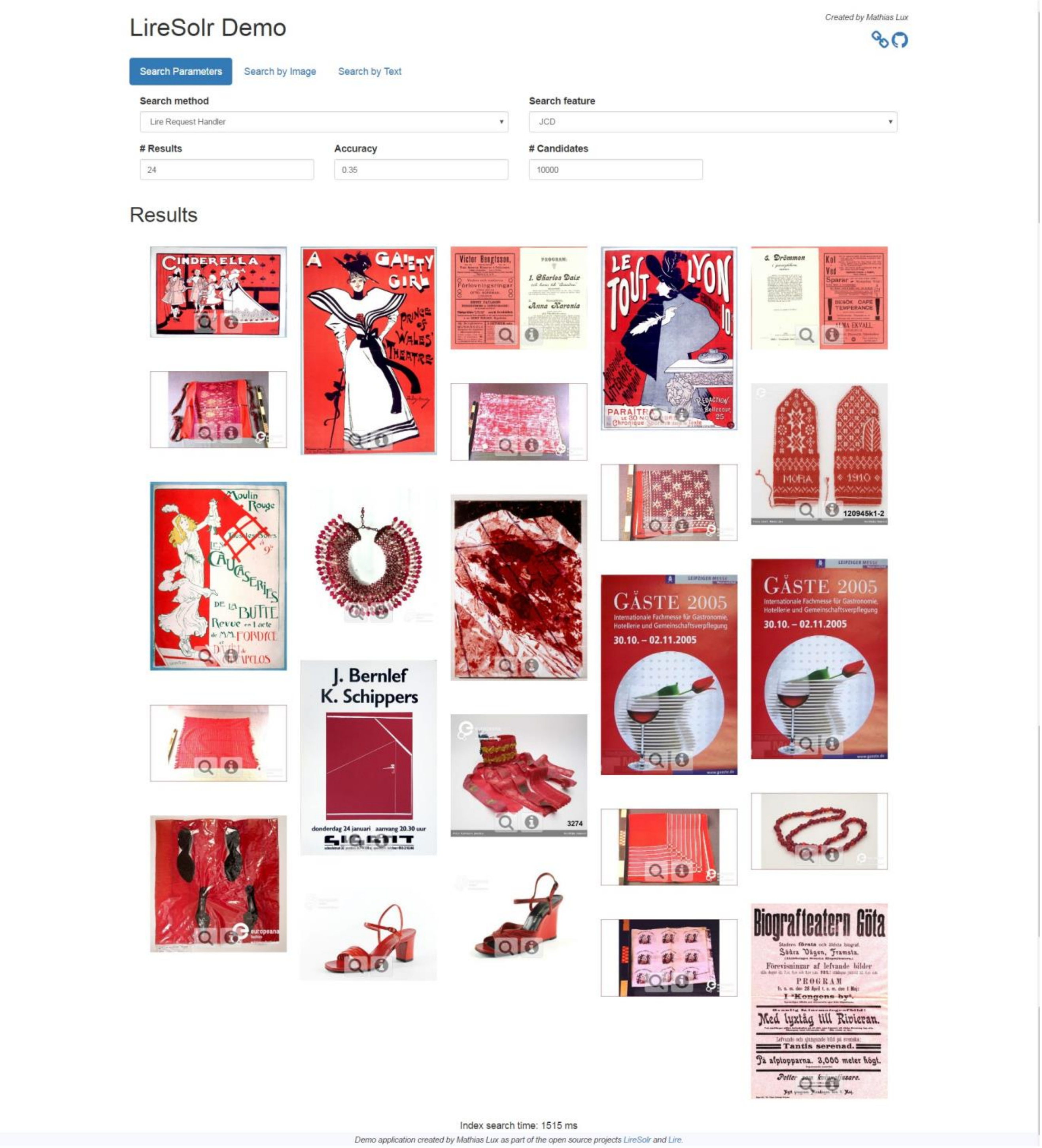


Figure 10: Counting the cases when an algorithm was indicated as being the best one, tailored by content categories

A second expert user was evaluating the service by assessing the performance over various themes and types of content. His evaluation criteria were synthesized as follows: “My ideal result would be an even mix of clearly similar (in shape or colour) images with some that are similar but in surprising or more subtle ways. And no obviously dissimilar images at all! My ideal is NOT for only near duplicates.” An example of good search results is presented in Figure 11, providing a good mix of quite similar and serendipitous items. The items in the results have a high variety of themes, but they provide a feeling of being a whole (i.e. a consistent view) when looking to the big picture.



LireSolr Demo

Created by Mathias Lux

Search method: Lire Request Handler

Search feature: JCD

Results: 24

Accuracy: 0.35

Candidates: 10000

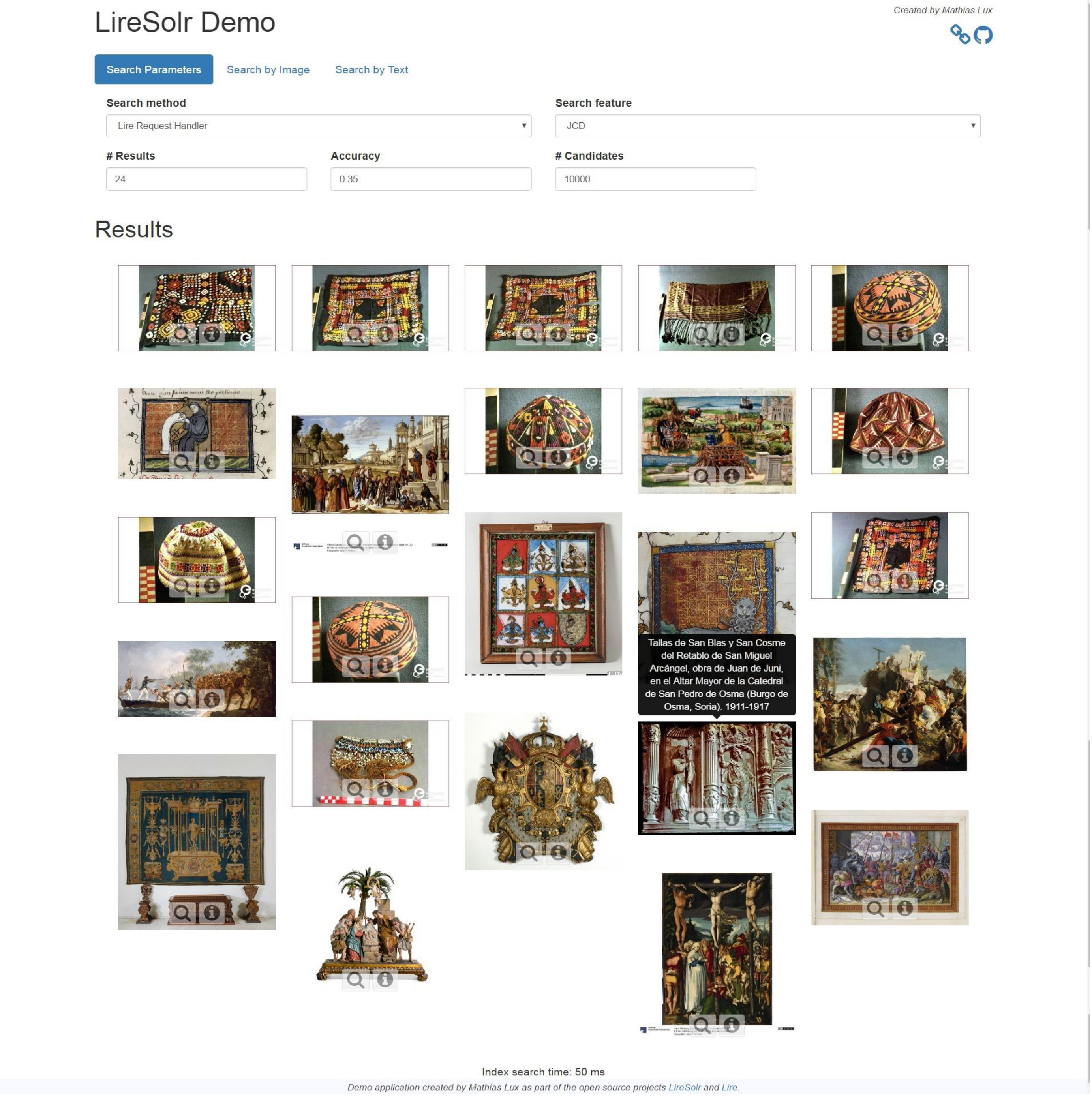
Results

Index search time: 1515 ms

Demo application created by Mathias Lux as part of the open source projects LireSolr and Lire.

Figure 11: An example of good search results

The assessment of best performing algorithms by both expert converges to a broad agreement. Despite of the fact that completely different search examples were used, the second expert also indicates that, on average, JCD is the best performing algorithm, illustrating it through some very good search results (see Figure 12 for example).



Search method: Lire Request Handler

Search feature: JCD

Results: 24

Accuracy: 0.35

Candidates: 10000

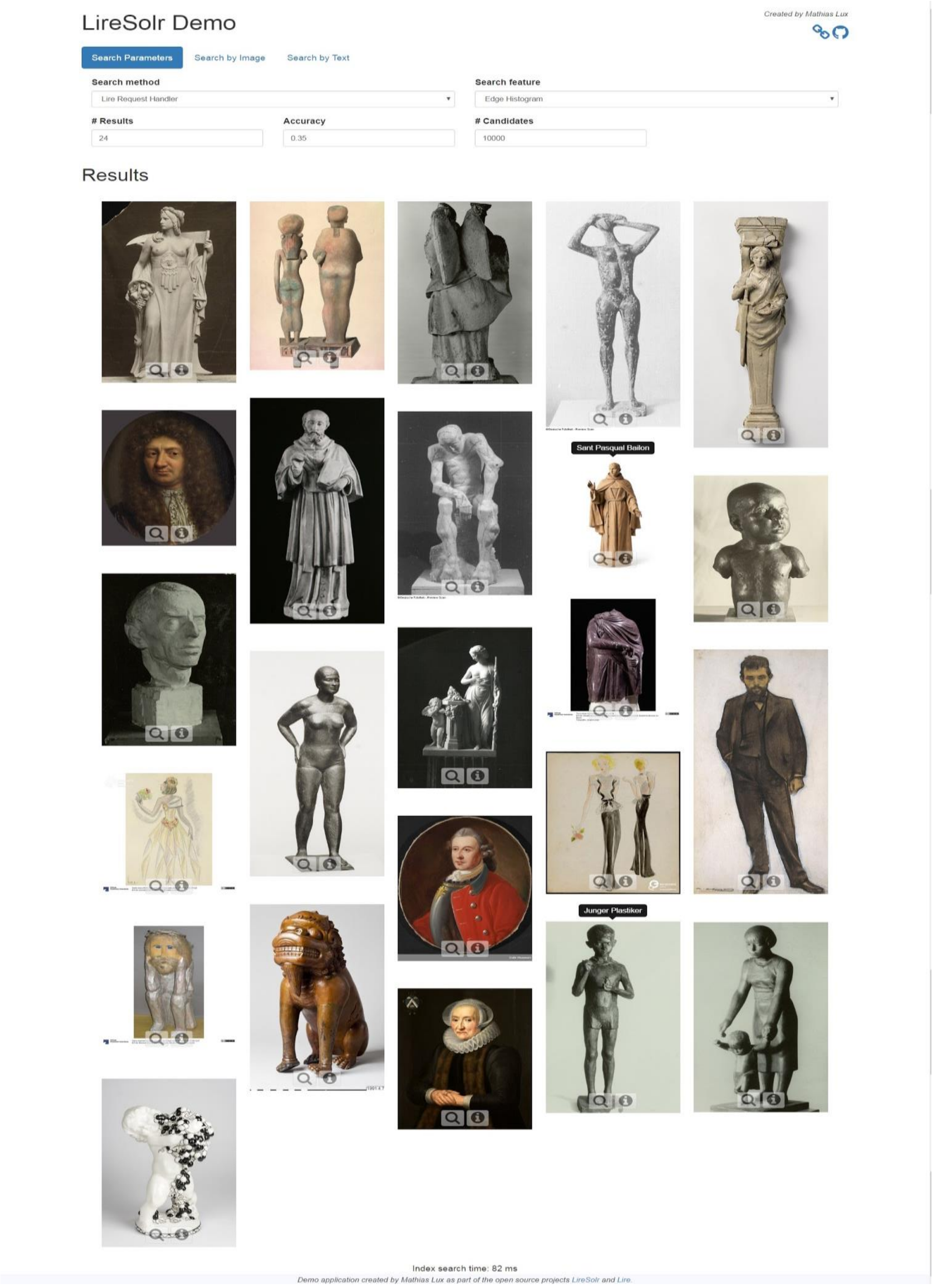
Tallas de San Blas y San Cosme del Retablo de San Miguel Arcángel, obra de Juan de Juni, en el Altar Mayor de la Catedral de San Pedro de Osma (Burgo de Osma, Soria). 1911-1917

Index search time: 50 ms

Demo application created by Mathias Lux as part of the open source projects LireSolr and Lire.

Figure 12: An example of JCD based search

The Edge Histogram was identified by the second expert to be best used for retrieval for “3D” objects like shoes, sculptures, small objects/carvings and jewellery (see Figure 13). PHOG is useful mainly for searching objects that are rich in texture but poor in color information, like ornamental objects or jewelries. In these cases PHOG may provide better results than the Edge Histogram and in some cases it provides good suggestions on paintings (i.e. as they are typically rich in information). However, for paintings the color information is an important component of the perceived similarity, and the colorless images that are often retrieved with PHOG are generally considered false positives[37].



Search method: Lire Request Handler

Search feature: Edge Histogram

Results: 24

Accuracy: 0.35

Candidates: 10000

Results

Index search time: 82 ms

Demo application created by Mathias Lux as part of the open source projects LireSolr and Lire.

[37] https://docs.google.com/presentation/d/1UN0AjqfjKcaHH0dMSzBFnNxNMP5PYdSYgmFiDkeeVrA/edit#slide=id.g1f822285fb_0_1

Figure 13: Search based on Edge Histogram using 3D object as query

User Evaluation

The expert evaluation provides good feedback on the overall performance of different algorithms, by using different types of search queries. It indicates that a combination of different features is likely to provide better search results, however this assessment is not enough for deriving weighting schemes that aim at improving the search performance. Also, the new created dataset doesn’t contain enough classification metadata to support an automated evaluation of alternative search algorithms.

Under these circumstances, an evaluation through the end users is required to address open issues of this first version of the image similarity API. The first goal of the experimental evaluation is to verify the correlation on the perceived similarity and quality of search results between the expert users and regular users. Additionally, we aim at collecting statistically relevant information that will be used to combine different image features in an algorithm that is able to outperform the currently available alternatives.

The experimental methodology was inspired by the results of expert evaluation, however, this was formalized in a more controlled experimentation setup and a less time demanding process is used[38]. The evaluation graphical user interface was created especially for this purpose (see Section Implementation of advanced image search services and demos), which allows an easier interaction and a more effective way for switching between alternative search algorithms. A lightweight spreadsheet[39] document was created to collect user feedback by indicating the best performing algorithm on a given query and the level of serendipity within the search results.

The user evaluation was carried out with 8 end users, evaluating a total of 75 search queries. For about a half of the search examples (50,6 %), at least one of the algorithms was able to provide good or very good results; while in another 24% of the cases were indicated acceptable search results. For 16 search queries (21%) none of the proposed search algorithms was able to provide satisfactory results, as the overall quality was indicated as being bad or very bad. The distribution of best performing algorithms tailored by overall quality of results is presented in Figure 14. This Figure does not include the distribution of the results in the case of non satisfactory search cases, as in this cases is not relevant anymore which algorithm provides the best results.
Within this evaluation, the Color Layout based search was indicated to perform the best in 18% of the cases, followed by Edge Histogram and PHOG, each of them being marked as the best in 14% of the cases. CEDD was assigned with a score of 12%. Surprisingly the JCD based algorithm was prefered in only 8% of the cases, however this algorithm was often mentioned as being the second best algorithm in the textual comments provided by users.
The differences between the evaluation results of the expert users and regular users, might be explained through different expectations when evaluating image similarities. However, as indicated in the previous sections, the distribution of the results is highly influenced by the type of the content selected for the query image. As the goal was to cover as many as possible positive and negative search examples, the users were asked to use an arbitrary criteria for choosing the images used for search. The selection of 75 search examples is not statistically representative for the different image types available in the dataset that includes 118k+ items. However, the overall

[38] https://docs.google.com/document/d/1zxoOkox-fgHTIZ2UKQ0j5AYmxszdL77-tzV2ArKWtIM/edit?ts=599846ad#
[39] https://docs.google.com/spreadsheets/d/1d2siTho_IZs6pEaSNqy_85Ft-IcEX0TyV3Kz1deIJa4/edit#gid=0

experimental results indicate that the image search functionality can be successfully integrated in Europeana Collections, possibly by replacing the “more like this” functionality for image content.

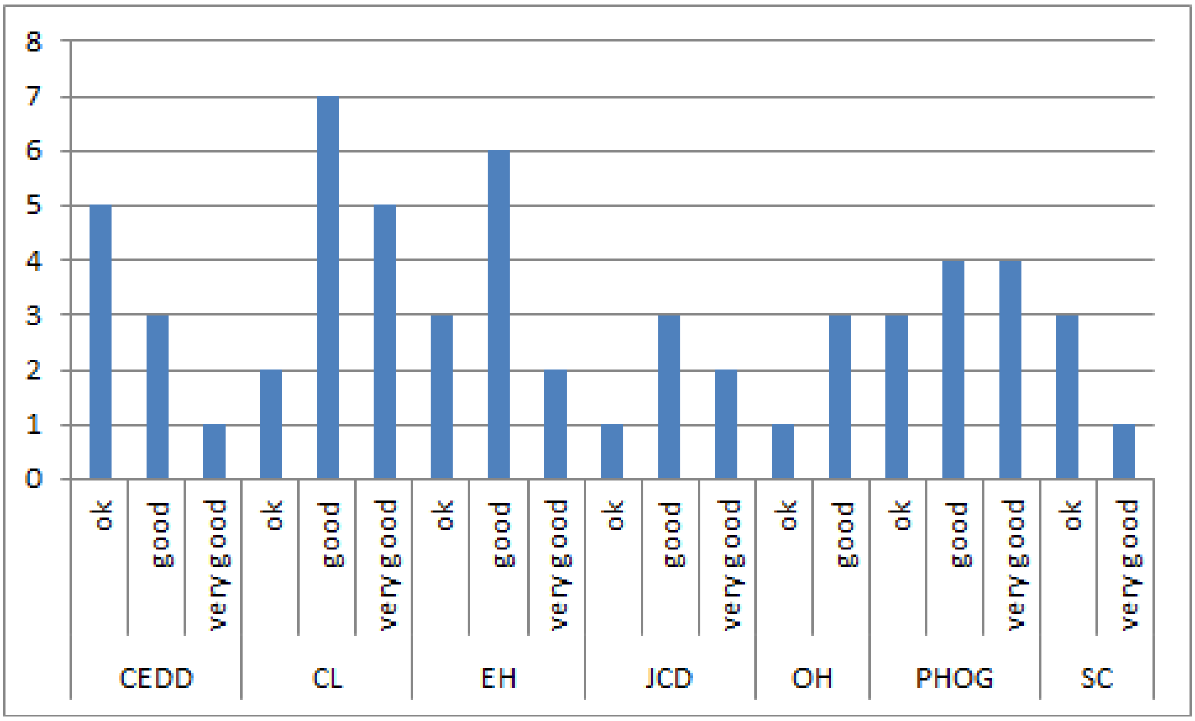


Figure 14: Counting the cases when an algorithm was indicated as being the best one, tailored by overall quality of results

Conclusions and Future Work

This deliverable presents the work carried out within the Subtask 6.3.4 having the goal develop advanced image search services for Europeana. For achieving this goal, we evaluated alternative state of the art solutions by assessing their search performance and their technical maturity. We were especially interested in the solutions compatible with the Europeana APIs technology stack and the ones that are able to ensure the level of scalability required by the size of the dataset and the average usage of Europeana APIs.

Concretely, we performed an evaluation of CDVS and LIRE based image search algorithms and concluded that LireSolr is complying with the technical requirements of the API, while several features (including JCD, CEED, PHOG and Edge Histogram) are already providing good search performance on the given dataset. CDVS based solution has the potential to improve the search accuracy for scenarios based on object identification or motif based retrieval. However the reference software implementation requires additional engineering work in order to make it production ready. A basic RESTful API for accessing the search functionality though HTTP request was implemented within the scope of this project, however this doesn’t provide yet the scalability required for production environments.

The development of the API and demo graphical user interfaces was carried out by following the initial activity planning presented in MS6.1. There was one significant deviation from the initial planning; as result of the technical evaluation of LireSolr library, it was decided to develop the API on top of it, rather than migrating the Europeana Creative search service to a Solr based implementation. Additionally, the enhancements developed into the LireSolr fork created some delays in the activity planning, which had the effect of postponing the evaluation activities for the last 2 months of the project.

The management of development activities was carried out by the APIs team, following the Europeana guidelines for API development, while technical infrastructure of Europeana was used for continuous integration and testing purposes. An initial assessment of how to integrate this new service into the collections was created, but the implementation is part of the future work. As future work on the technical development we foresee activities that aim at integrating the CDVS based search in a Solr based solution. Based on the user evaluation results, an algorithm that combines several image features is expected to be implemented for the beta release of the service.

References

[Amato 2011] Giuseppe Amato, Paolo Bolettieri, Fabrizio Falchi, Claudio Gennaro, and Fausto Rabitti, “Combining local and global visual feature similarity using a text search engine” in CBMI, 2011, pp. 49–54.

[Gordea 2015] Sergiu Gordea and Michela Vignoli, “CultureCam: Visual exploration of cultural heritage content by professional designers” in 2015 IEEE International Conference on Multimedia & Expo Workshops (ICMEW), 2015, pp. 1-6

[Sikora 2001] T. Sikora, “The mpeg-7 visual standard for content description-an overview,” Circuits and Systems for Video Technology, IEEE Transactions on, vol. 11, no. 6, pp. 696–702, Jun 2001.

[Duan 2015] L. Duan, T. Huang, W. Gao, "Overview of the MPEG CDVS standard", *Proc. IEEE Data Compression Conf.*, pp. 323-332, 2015-Apr.

[Bianco 2015] S. Bianco, D. Mazzini, D.P. Pau, R. Schettini, Local detectors and compact descriptors for visual search: A quantitative comparison, Digital Signal Processing, Volume 44, September 2015, Pages 1-13

[Lux 2008] Lux Mathias, Savvas A. Chatzichristofis. Lire: Lucene Image Retrieval – An Extensible Java CBIR Library. In proceedings of the 16th ACM International Conference on Multimedia, pp. 1085-1088, Vancouver, Canada, 2008

[Rublee 2011] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. 2011. ORB: An efficient alternative to SIFT or SURF. In *Proceedings of the 2011 International Conference on Computer Vision* (ICCV '11). IEEE Computer Society, Washington, DC, USA, 2564-2571.

[Alcantarilla 2012] Alcantarilla, P. F., Bartoli, A., Davison, A. J.: KAZE features. Proceedings of the 12th European conference on Computer Vision, 214{227 (2012)

[Lux et al. 2016] Mathias Lux , Michael Riegler , Pål Halvorsen , Konstantin Pogorelov , Nektarios Anagnostopoulos, LIRE: open source visual information retrieval, Proceedings of the 7th International Conference on Multimedia Systems, May 10-13, 2016, Klagenfurt, Austria [doi>10.1145/2910017.2910630]

[Amato 2002] "Approximate similarity search in Metric spaces", G. Amato, Computer Science Department, University of Dortmund, June 2002

[Hill 2016] “Searching for Inspiration”: User Needs and Search Architecture in Europeana Collections, *Timothy David Hill, Valentine Charles, Antoine Isaac, Juliane Stiller,* Europeana Foundation, United Kingdom, Proceedings of the 79th ASIS&T Annual Meeting, vol. 53, 2016, https://www.asist.org/files/meetings/am16/proceedings/submissions/papers/17paper.pdf

Annex 1 - Activity Plan

Activity	Short Description	Due date
Dataset Aggregation	Selection of image content used by the search service	M7
Migration of existing image search service	Implementation of existing image search service using Solr (including feature extraction)	M8
Implementation of the image search API	Implementation of Restful API for existing image search service (Search API)	M8
Filtering and faceting	Implementing filtering and faceting functionality in the search API	M9
Experimental evaluation	Evaluate effectiveness of novel search algorithms on the given dataset	M10
Migration of novel image search algorithms	Implement novel algorithms using solr based solution (including feature extraction)	M12
Integration in Europeana Collections	Develop GUI components for integrating image search in Europeana Collections	M12
Final version of the search API	The final deployment of the image search service.	M13

Annex 2 - Content Selection for Evaluation Dataset

Art subsets

Paintings

The paintings contained within the thematic art collections.

Link to search result with a selection of paintings.

Sculpture

The sculptures contained within the thematic art collections.

Link to search result with a selection of sculptures.

Note: There may well be drawings and other indirect images of sculptures in this result set. It may be difficult to narrow it down to e.g. only photographs of sculptures.

Posters

The posters contained within the thematic art collections.

Link to a search result with a good selection of posters. Note: NOT all posters in Europeana.

Illuminated manuscripts

The illuminated manuscripts contained within the thematic art collections.

Link to search results for illuminated manuscripts.

Tapestries

The tapestries contained within the thematic art collections.

Link to search results with tapestries

Porcelain ceramics

Link to a search results for porcelain ceramics (API-query)

Fashion subsets

Shoes

Link to search results for shoes (API-query)

Headgear

Link to search results for headgear (API-query)

Fashion shows

The fashion photography contained within the thematic fashion collections.

what:"fashion show" OR "fashion photograph"

Fashion sketches, drawings and illustrations

Link to search results for fashion sketches, drawings and illustrations (API-query)

Jewellery

The jewellery contained within the thematic fashion and art collections.

A good set within the Fashion Collections.

A set of of jewellery within the Art Collections

Objects and tools

Music instruments

The music instruments contained within the thematic music collections.

Link to images of music instruments