Semantic Web 1 (2021) 1–5
IOS Press

Linked Open Images: Visual Similarity for the Semantic Web

Lukas Klic [a,*],
[a] *I Tatti, The Harvard University Center for Italian Renaissance Studies, Harvard University, Florence, Italy*
E-mail: lklic@itatti.harvard.edu

Abstract. This paper presents ArtVision, a Semantic Web application that integrates computer vision APIs with the ResearchSpace platform, allowing for the matching of similar artworks and photographs across cultural heritage image collections. The field of Digital Art History stands to benefit a great deal from computer vision, as numerous projects have already made good progress in tackling issues of visual similarity, artwork classification, style detection, gesture analysis, among others. Pharos, the International Consortium of Photo Archives, is building its platform using the ResearchSpace knowledge system, an open-source semantic web platform that allows heritage institutions to publish and enrich collections as Linked Open Data through the CIDOC-CRM, and other ontologies. Using the images and artwork data of Pharos collections, this paper outlines the methodologies used to integrate visual similarity data from a number of computer vision APIs, allowing users to discover similar artworks and generate canonical URIs for each artwork.

Keywords: Linked Open Data, Computer Vision, Cultural Heritage, Semantic Web, Visual Similarity

1. Introduction

Computer Vision (CV) for cultural heritage, in particular when applied to two-dimensional artworks, has been a topic of growing interest to scholars and institutions. Various conferences, symposia, and workshops have been organized over the years to explore the usefulness of this technology. While these tools have made much more headway in commercial sectors, the cultural heritage domain has seen less benefit as most CV APIs are generally built with images of real-world objects rather than artworks and what they represent. While this paper provides a report on some CV APIs and their applicability to the cultural heritage domain, its main focus is to provide a report on a Semantic Web application that makes these tools available to non-technical users using the Pharos image collections.

Pharos[1], the international consortium of photo archives, is a collaboration between fourteen European and North American art historical research institutes working to create an open and freely accessible digital research platform to provide consolidated access to their collections of photographs and associated scholarly documentation. The institutions collectively own an estimated 25 million photographs documenting works of art and architecture and the history of photography itself, forming the largest repository of images for the field of art history. This work is supported by a grant from the Andrew W. Mellon foundation[2] to work on the first phase of the project, using the ResearchSpace platform, a Semantic Web knowledge representation research environment created and developed by the British Museum to support a move away from traditional static and narrow data indexes to dynamic and richer contextualized information patterns [1]. The Pharos collections are published using the CIDOC-CRM ontology, the methodology of which will be outlined in forthcoming publications following the publication of the platform in 2022.

*Corresponding author. E-mail: lklic@itatti.harvard.edu.
[1] "PHAROS: The International Consortium of Photo Archives." Accessed March 15, 2021. http://pharosartresearch.org/
[2] https://mellon.org/grants/grants-database/grants/the-frick-collection/1811-06391/

The ArtVision platform, published at https://vision.artresearch.net, provides a user interface for institutional staff from participating Pharos institutions to log in and match up artworks across collections. By matching up artworks, the platform generates a canonical, dereferenceable URI for each artwork, allowing these to be reused across the web and in scholarly publications. Through SPARQL federation across REST APIs, the application also allows users to upload an image to the platform and find visually similar artworks in the Pharos collections in real-time.

2. Background

Art historians have a long tradition of writing about images, one could say, transcribing a visual language to a textual one. As with all translation activities, there is a loss of meaning in this process. While the perception of images is instantaneous and universal, writing is bound by time and cultural bias [2]. Computer vision has emerged as a powerful tool for the field of Digital Art History with the potential to bridge some of these gaps between images and text, allowing images to dialog with one another without the mediation or dependence on texts. A great deal of experimentation has already been done in applying CV to images of artworks: attempts to automatically classify paintings [3], recognize style [4], object detection studies to identify objects in artworks [5], evaluating influence [6], and large-scale analysis of broad concepts within artworks such as gesture [7]. The opportunities offered by this technology are numerous but the solutions are generally ad-hoc and are rarely made available to non-technical users. While the analytical capacity of CV generally cannot surpass that of an art historian, the ability to perform specific tasks at a scale otherwise unattainable by humans makes it attractive to certain use-cases for art historians [8].

The matching of artwork images across collections and comparing metadata at scale has also been experimented with by various projects[3] [9]. This application builds on and extends some of this work by integrating it with semantic technologies. ArtVision not only provides a user interface to end-users, but also makes the data openly available for interpretation and reuse through a SPARQL endpoint. For institutions looking to link digital collections of images that document artworks across the web of data, the platform provides programmatic access to both its CV APIs as well as existing similarity data between images. These data can then be used to build new machine learning models by providing training data to refine notions of visual similarity for artworks. Additionally, institutions and researchers can programmatically obtain canonical URIs for visually similar artworks, which can then be integrated into their own collection platforms for discovery by end-users.

3. Visual Search for Cultural Heritage

In order to develop the ArtVision platform, the author tested a range of publicly available Computer Vision and Machine Learning APIs for their ability to find visually similar artworks. In order to gauge their usefulness, each tool was tested with specific use-cases. As outlined in the Table 1, three forms of visual search were evaluated: exact image match, visually similar, and partial image match. Image labels or the tagging of images was beyond the scope of this testing but was noted in order to allow for a future expansion of the platform to support this functionality.

An exact image match, meaning that two images are nearly identical, is computationally a more trivial task than visually similar images or those that allow for partial image matching. In this case, it is assumed that the crop and content of images would be nearly identical, with only minor variations. Many tools can calculate this kind of similarity, but the implementation that was the most robust for this use case was Match[4], a reverse image search based on ascribe/image-match[5]. This API provides a simple signature for each image which is stored in ElasticSearch for quick retrieval. It scales very well to billions of images, a functionality that few other tools can claim. The use case for a near or exact image search is limited since different images of artworks will almost always subsume some level of variation in a crop, color, or angle of photograph. Nevertheless, these results can provide an additional

[3]John Resig - Building an Art History Database Using Computer Vision. https://johnresig.com/blog/building-art-history-database-computer-vision/. Accessed 10 Mar. 2019.

[4]Crystal ball: Scalable Reverse Image Search Built on Kubernetes and Elasticsearch: Dsys/Match. 2016. Distributed Systems, 2019. GitHub, https://github.com/dsys/match

[5]EdjoLabs/Image-Match: Quickly Search over Billions of Images. https://github.com/EdjoLabs/image-match. Accessed 10 Mar. 2019.

Table 1
CV / ML tools tested for visual search

	exact image match	visually similar	partial image match	custom image index	image labels/tags	custom classifiers
Google Cloud Vision		x			x	
Amazon Rekognition					x	
Clarifai	x	x		x	x	x
Pastec.io	x	x	x	x		
Match	x			x		
InceptionV3	x	x		x	x	x
IBM Watson					x	x
Microsoft Computer Vision API					x	x
Cloudsight					x	

layer of verification to other tools that offer a more "fuzzy" similarity matching. It can also be useful to programmatically find historical photographs that are prints made from the same negative, or multiple scans of the same photograph.

APIs that support searches for images that have varying degrees of "similarity" were found to be more challenging. Earlier tests by John Resig[6] on behalf of the Pharos consortium showed that Pastec.io[7], an open-source image similarity search API, can provide a very usable set of results for historical photographs.

Results provided by Google Cloud Vision, Amazon Rekognition, IBM Watson, Microsoft, and Cloudsight were not usable for the ArtVision use-case as they only allowed image tagging, or only allowed for the comparison of images to ones available on the web. Tools that were not tested were not openly accessible through an API (free or commercial) or did not allow for searching within a set of images provided by the user. These include Visual Search by Machine Box[8], Deep Video Analytics [9], Bing Visual Search API[10], the Replica project [11], and ArtPi [12].

Among the APIs or models that did allow for an internal index of images, Clarifai and InceptionV3 provided the least useful results, likely because they were trained on images of real world objects and use Convolutional Neural Networks rather than the bag-of-words methodology [10] employed by Pastec or the pHash[13] library used by Match.

The Pastec API was able to produce the most useful results that were applicable to the ArtVision use-case, and was able find identical and visually similar artworks even where the images were significantly different from one another. Matches were found from different images even when the frame was removed, when the crop was different, where one image was in grayscale and another in color, with variations in the angle in which the photograph was taken, or when one artwork was a copy of another.

4. System Architecture

The ArtVision platform was built using a separate instance of ResearchSpace platform from the main Pharos instance, and all data was stored in its Blazegraph database backend. Most CV APIs require that images be added to an internal index through a REST API. Once images are added, the internal index can be queried again with a single image to return all similar images, along with their visual similarity scores, as defined by that specific API. Since adding a large number of images to a CV index can be time-consuming, the process of adding images cannot be performed in real-time by ResearchSpace. For this reason, a separate

[6] John Resig - Italian Art Computer Vision Analysis. https://johnresig.com/research/italian-art-computer-vision-analysis/. Accessed 10 Mar. 2019.

[7] Pastec, the Open Source Image Recognition Technology for Your Mobile Apps. http://pastec.io/. Accessed 10 Mar. 2019.

[8] Hernandez, David. "Visual Search by Machine Box." Machine Box, 7 Oct. 2017,https://blog.machinebox.io/visual-search-by-machine-box-eb30062d8abe.

[9] Deep Video Analytics. https://www.deepvideoanalytics.com/ Akshay Bhat from Cornell University. Accessed 10 Mar. 2019.

[10] Bing Visual Search Developer Platform. https://www.bingvisualsearch.com/docs. Accessed 10 Mar. 2019.

[11] Diamond. https://diamond.timemachine.eu/. Accessed 10 Mar. 2019.

[12] ArtPi - Artrendex. http://www.artrendex.com/artpi Accessed 10 Mar. 2019.

[13] http://www.phash.org/

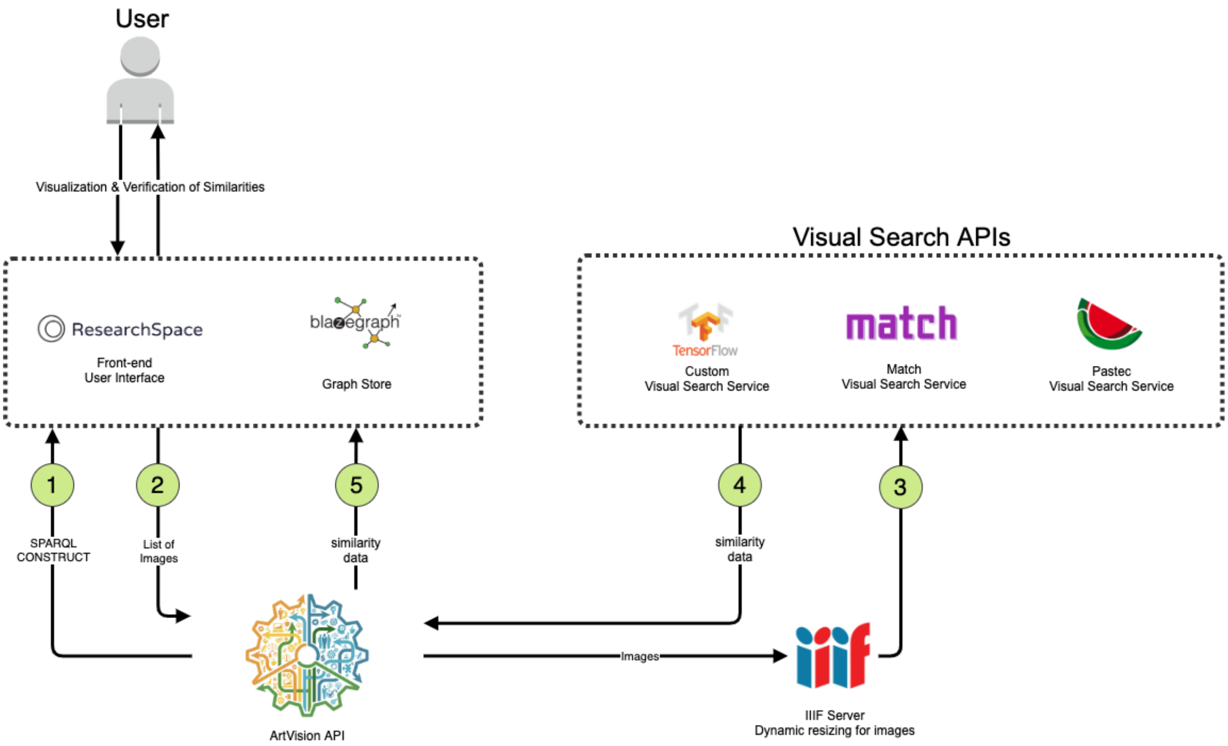

Fig. 1. Software architecture for the ArtVision API

ArtVision API[14] was developed to handle the process of downloading the image URLs from the main Pharos endpoint through a SPARQL construct query. In order to alleviate timeout issues associated with retrieving a large number of results from the Pharos endpoint, the ArtVision API runs on the same server as the Pharos endpoint, and queries Blazegraph directly through its own internal API. Built as a Java application, the architecture is agnostic to the visual search tool being used, allowing for the system to grow with additional services over time.

As seen in Figure 1, the ArtVision API first performs a SPARQL construct query(1) against the Pharos endpoint, which returns a list of images(2) and saves them according to a predefined data model in a turtle file. It then iterates through this list of images one at a time and sends them to each CV API (3) for indexing after resizing them through the IIIF API[15]. After indexing, another query is sent to retrieve similar images(4), the results of which are transformed according to the predefined data model and inserted via SPARQL(5) back into the Blazegraph database. When an image is indexed by a specific CV API, the identifier within that index is also materialized to the image URI node within the Pharos dataset, ensuring that when the initial SPARQL construct query is re-run, that image does not get re-indexed to any particular API. Each CV API is added through a configuration file, and where necessary, java classes can be added or modified to handle different API parameters. The data model and SPARQL queries are also completely configurable to serve a wide range of use-cases. As all data are stored in the graph database, the ResearchSpace templating engine provides a user interface to these data, allowing users to review results and perform additional actions.

[14]The ArtVision API architecture was designed by Lukas Klic and implemented by Manolis Fragiadoulakis of SmartUp Data Solutions. https://github.com/ArtResearch/vision-api

[15]https://iiif.io/api/image/3.0/

5. Data Model

The underlying data model was instrumental in ensuring the interoperability and extensibility of similarity results from multiple CV APIs. At the most basic level, the requirement was to describe a level of

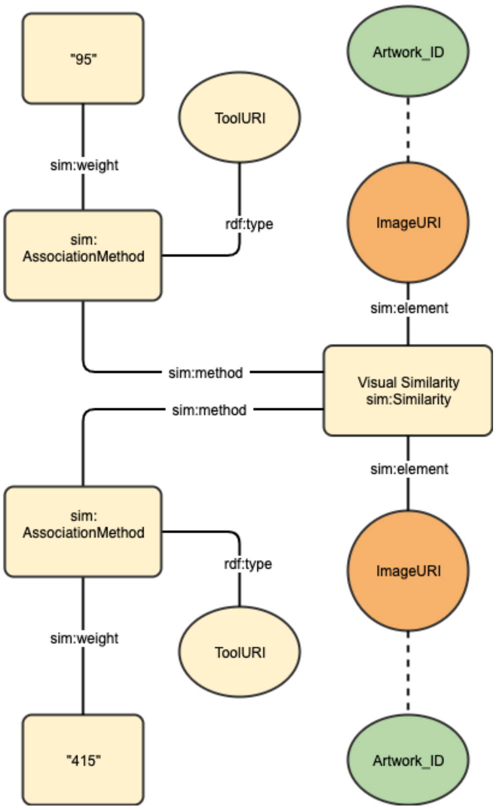

Fig. 2. ArtVision data model

similarity between two target images according to a specific methodology (or API) and have a similarity score for that methodology. Similarity is always bidirectional and a search for one image in pair should generally yield the same score as searching for the other.

As shown in Figure 2, the Similarity Ontology[16], originally designed for music but applicable to any domain, provided the necessary ontological framework to encode these similarities for any pair of images. A single similarity node for each image pair can have various association methods, each with its own weight. Although the website documenting the ontology is no longer available (it was last modified in 2010), the internet archive has a copy and the ontology itself is available on Linked Open Vocabularies[17].

[16] https://lov.linkeddata.es/dataset/lov/vocabs/sim

[17] Linked Open Vocabularies (LOV). https://lov.linkeddata.es/dataset/lov/vocabs/sim. Accessed 10 Mar. 2019

6. URI Hashing

One challenge with having a single similarity node connecting each pair of images was generating this URI programmatically without having to perform a lookup to see if one already existed. In order to allow for additional CV APIs to be added at a later date, the URI of the similarity node would need to be generated as a hash based on the URI of the two images that are connected to that node. The ability to calculate the URI of this node through SPARQL as well as within the ArtVision API was also a requirement in order to allow for the creation of these nodes through the ResearchSpace infrastructure.

Fig. 3. URI hashing for ArtVision

As seen in Figure 3, the similarity node should be the sum of a hash of the two images, so that the hash of Image 1 + Image 2, as well as the inverse (Image 2 + Image 1), would equal that of the similarity node. This hashing was implemented by obtaining a SHA1 hash of each image and removing all characters that are not integers through a regular expression, and adding them together.

Figure 4 illustrates how this hashing can be calculated through SPARQL, allowing for this calculation to be implemented both in the separate java application of the ArtVision API as well as through ResearchSpace via SPARQL. Depending on the number of images in the dataset, other hashing functions such as SHA256 or SHA512 could also be implemented in order to avoid the possibility of clashing URIs.

7. SPARQL federation across REST APIs

In addition to the ArtVision API that was developed as a separate application, a SPARQL to REST API federation service was developed within ResearchSpace that would allow end-users to upload their own images. This visual search service allows users to drag-and-drop an image onto the platform or paste an image URL into a webform. This federation service was developed by researchers at the Harvard Digital Humani-

```
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
SELECT ?image1 ?image2 ?imageId1 ?imageId2 ?imagePairHash

WHERE {
  BIND( "https://example.org/image1.jpg" AS ?image1) .
  BIND( "https://example.org/image2.jpg" AS ?image2) .
  BIND ( REPLACE(SHA1(?image1), "[A-Z]+" , "", "i") AS ?imageId1).
  BIND ( REPLACE(SHA1(?image2), "[A-Z]+" , "", "i") AS ?imageId2).
  BIND ( xsd:integer(?imageId1) + xsd:integer(?imageId2) AS ?imagePairHash)
}
```

Fig. 4. SPARQL URI hashing for ArtVision

ties Lab[18] with help from the ResearchSpace team and was generalized to allow for the querying of a wide range of REST endpoints. Leveraging the RDF4J Storage and Inference Layer (SAIL) libraries, this integration allows for the real-time querying of any REST endpoint. By implementing this federation service, each CV API is registered as a SPARQL repository and mapping files provide a translation between the API results using a predefined ontology. This functionality allowed for a Google image search type lookup against the Pharos endpoint, allowing users to find visually similar images in the Pharos collections.

Users can enter the URL of an existing image and visualize matching results in the Pharos dataset. This functionality is particularly useful for art historians who want to find the corresponding record in the Pharos dataset when they are unable to find the record through a text search, or when they want to simply find images that are visually similar to the artwork they are researching.

8. ArtVision User Interface

The ArtVision ResearchSpace platform also allows institutional staff to log into the platform and match up artwork records across institutions. Two user interfaces were developed to support this functionality.

In a more simplified view as illustrated in figure 5, staff can see a pair of visually similar images and a limited subset of metadata to quickly compare two artworks. By clicking the "same" button, they create owl:sameAs links between these two artwork records that will result in them being displayed as a single record in the Pharos platform. In the Pharos dataset,

[18]Gianmarco Spinaci, https://github.com/researchspace/researchspace/pull/187

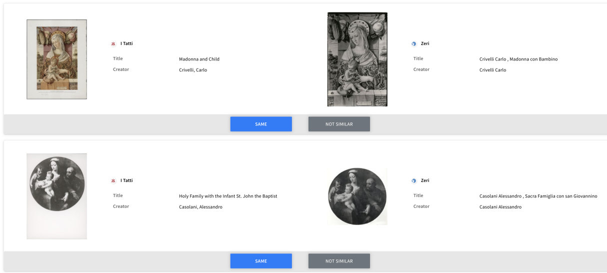

Fig. 5. Simplified view: comparing images and metadata from Pharos collections

each artwork as cataloged by a particular institution is stored in a named graph that allows for the tracking of the provenance of these data. Each artwork is represented by both an institutional URI as well as a canonical Pharos URI. By clicking the "same" button in the ArtVision platform, one of the Pharos URIs is marked as being deprecated and owl:sameAs links are created between the two institutional URIs and the remaining Pharos URI.

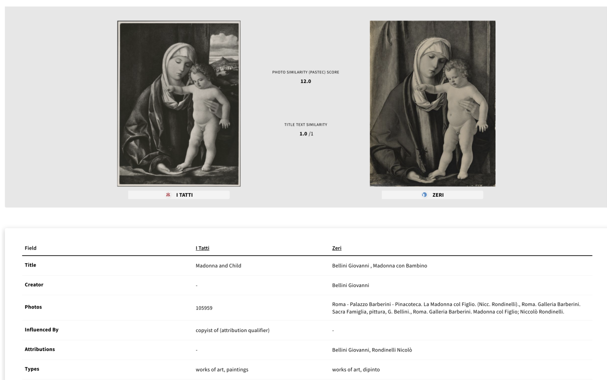

Fig. 6. Detailed view: comparing images and metadata from Pharos collections

A second interface as shown in figure 6, allows staff to view both artwork records side-by-side and more of the metadata for these records. Clicking on an image, both images are shown in a IIIF viewer that allows users to zoom in and perform an in-depth comparison of these two images. Visual similarity scores for each CV API are shown between the two images, allowing users to visualize the raw results from multiple CV APIs. This interface also allows for users to make assertions about different types of similarity using the CIDOC-CRM Inf Argumentation Model as implemented in the ResearchSpace platform. Here, the similarity is not bidirectional as one image may be a copy of another, be after restoration, or the two images could be derived from the same photographic negative.

9. Visual Cataloging

Fig. 7. Matching iconographic themes

Another use-case that could prove to be disruptive to libraries and archives needing to catalog images of artworks, is to leverage the functionality of partial image match or similar image matching to assist in the cataloging of images in batch. As seen in Figure 7 the ability to use images to search for others with the same iconographic theme presents a powerful tool for institutions to be able to apply metadata to vast numbers of images. This is particularly useful in situations where image metadata are not available and could embolden the publishing efforts of historic photograph collections by institutions that have no metadata.

Partial image match also allows for cropped images or details of works to be found within images that contain the whole. There are numerous use cases for this kind of functionality, especially with images that are cropped, or images of artworks that have been split up into pieces, as is often the case with altarpieces and triptychs.

Fig. 8. Matching details of artworks to their whole

For this use-case, ArtVision is once again able to return meaningful results. As seen in figure 8, a small detail of an artwork was matched with the Pastec API. With partial image match functionality, visual cataloging can also be implemented to search the verso of photographs as well.

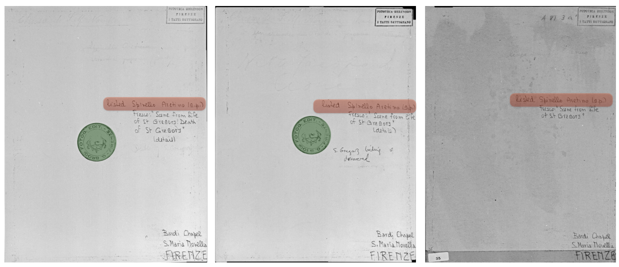

Fig. 9. Matching stamps and handwritten text on the verso of photographs

As illustrated by figure 9, this functionality could also be implemented with photographer or institution stamps on the verso of photographs. These kinds of searches could also be very useful for scholars doing research on the history of photography, tracking the works of particular photographers across collections that have not published these metadata as part of their datasets. Visual cataloging can be implemented by allowing users to leverage the IIIF API to select a portion of an image and query the CV API to retrieve images with similar details. Through the ResearchSpace platform, users could then select groups of images to which they would then like to apply metadata in batches.

10. Impact and Future Work

The ArtVision platform can also be used to programmatically find incongruencies in metadata. A single artwork that was documented in two different photographs may have been cataloged as having two separate artists with different production dates. This is a fairly common cataloging mistake with images of artworks that are lost, where the only data a cataloger has to work with is the photograph itself. Using visual similarity search, data about lost artworks can be easily merged and these histories could be reconstructed.

The visual matching of artworks also allows the Pharos platform to generate canonical URIs for any artwork within the collection. While many vocabularies are available to disambiguate and provide identifiers for artists (ULAN[19]), places (TGN[20], GeoNames[21]), Institutions (VIAF[22]), and bibliographic entities (OCLC[23]) across the web, the challenge of providing identifiers to artworks has never been properly tackled. The Getty CONA vocabulary[24] has made some efforts on this front but has not seen much development over the last ten years [11]. ArtVision seeks to overcome these challenges by serving as a linking mechanism between institutions that are owners of artworks, institutions that have reproductions of these artworks, and scholars who seek to reference them through Linked Data services.

In addition to providing artwork disambiguation services to individuals and institutions seeking to match up images of artworks across the web, ArtVision enables the linking of visually similar artworks, copies, restorations, preparatory drawings and their subsequent works, cropped images of larger artworks, or two photographic prints that have been produced from the same negative. The platform aims to democratize the accessibility of CV functionality to non-technical users. Aggregating similarity data generated by CV APIs together with data from matched artworks performed through a manual process will also enable the creation of Machine Learning models that are more fine-tuned to a specific research need. As the dataset of similarity data on artworks grows, new classification models can be trained to refine various notions of visual similarity.

The platform will also serve as an attractive tool for institutions that need to digitize image collections but do not have the capacity to catalog them, as is the case with many Pharos partners. Over time, image collections from museums and archives can be integrated, allowing for the linking of archival records to museum collection websites across the web. The platform could also prove to be transformative for scholars who typically need to search through collections data from multiple repositories. These data would otherwise be spread out in various silos across the web, with a limited ability to track them down if not with visual search. While Google image search has been useful to scholars when searching for copies of similar images, most images from institutional repositories are not indexed [12].

Although the ArtVision platform was developed specifically to address the needs of institutions that publish data and images about artworks, the methodology is by no means restricted to this field. The open-source software could be reused by heritage institutions to match images across manuscript collections or historical documents that have some type of visual similarity. The broader scientific community could also stand to benefit where there is a need to match images in other domains, as well as provide a publicly accessible SPARQL endpoint to provide unique identifiers for these entities. As many artworks are lacking in titles and unique identifiers, visual search has proven to be the only way to reconcile these records across collections. The platform has already made a substantial impact on the publishing efforts of the Pharos consortium, automating a task that would otherwise be impossible for humans. By exposing the APIs publicly and allowing non-technical users to perform visual searches across Pharos collections, ArtVision facilitates collection interoperability and the cross-pollination of collections data, disrupting barriers posed by proprietary databases where information is kept in silos. It opens new doors to performing large-scale analysis on artworks, providing unprecedented opportunities to individuals, institutions and the field of Digital Art History more broadly.

Acknowledgements

The ArtVision Java API was implemented by Manolis Fragiadoulakis, and the ResearchSpace user in-

[19] https://www.getty.edu/research/tools/vocabularies/ulan/
[20] https://www.getty.edu/research/tools/vocabularies/tgn/index.html
[21] https://www.geonames.org/
[22] http://viaf.org/
[23] https://www.worldcat.org/
[24] Cultural Objects Name Authority (Getty Research Institute). http://www.getty.edu/research/tools/vocabularies/cona/. Accessed 10 Mar. 2019.

terface for ArtVision was implemented by Andreas Michelakis together with Nikos Minadakis of SmartUp Data Solutions. A special thanks to Gianmarco Spinaci and Remo Grillo of the Harvard Digital Humanities Lab for their contributions, Dominic Oldman and Artem Kozlov from the British Museum, Louisa Wood Ruby of the Frick Collection, and the Andrew W. Mellon Foundation for supporting this work.

References

[1] D. Oldman, D. Tanase and S. Santschi, The problem of distance in digital art history: A ResearchSpace case study on sequencing Hokusai print impressions to form a human curated network of knowledge, *International Journal for Digital Art History* (2019), 5.29–5.45. doi:10.11588/dah.2019.4.72071. https://journals.ub.uni-heidelberg.de/index.php/dah/article/view/72071.

[2] J. Elkins and M. Naef, *What is an image?*, University Park, Pa., 2011. ISBN 978-0-271-05064-5.

[3] W.R. Tan, C.S. Chan, H.E. Aguirre and K. Tanaka, Ceci n'est pas une pipe: A deep convolutional network for fine-art paintings classification, in: *2016 IEEE International Conference on Image Processing (ICIP)*, 2016, pp. 3703–3707. doi:10.1109/ICIP.2016.7533051.

[4] S. Karayev, M. Trentacoste, H. Han, A. Agarwala, T. Darrell, A. Hertzmann and H. Winnemoeller, Recognizing Image Style, *arXiv:1311.3715 [cs]* (2013), arXiv: 1311.3715. http://arxiv.org/abs/1311.3715.

[5] E.J. Crowley and A. Zisserman, In Search of Art, in: *Computer Vision - ECCV 2014 Workshops*, L. Agapito, M.M. Bronstein and C. Rother, eds, Lecture Notes in Computer Science, Springer International Publishing, Cham, 2015, pp. 54–70. ISBN 978-3-319-16178-5. doi:10.1007/978-3-319-16178-5_4.

[6] A. Elgammal and B. Saleh, Quantifying Creativity in Art Networks, *arXiv:1506.00711 [cs]* (2015), arXiv: 1506.00711. http://arxiv.org/abs/1506.00711.

[7] L. Impett and S. Süsstrunk, Pose and Pathosformel in Aby Warburg's Bilderatlas, in: *Computer Vision – ECCV 2016 Workshops*, Vol. 9913, G. Hua and H. Jégou, eds, Springer International Publishing, Cham, 2016, pp. 888–902, Series Title: Lecture Notes in Computer Science. ISBN 978-3-319-46603-3 978-3-319-46604-0. doi:10.1007/978-3-319-46604-$0_6$1.

[8] L. Manovich, Computer vision, human senses, and language of art, *AI & SOCIETY* (2020). doi:10.1007/s00146-020-01094-9.

[9] B. Seguin, The Replica Project: Building a visual search engine for art historians, *XRDS: Crossroads, The ACM Magazine for Students* **24**(3) (2018), 24–29. doi:10.1145/3186653.

[10] J. Yang, Y.-G. Jiang, A.G. Hauptmann and C.-W. Ngo, Evaluating bag-of-visual-words representations in scene classification, in: *Proceedings of the international workshop on Workshop on multimedia information retrieval*, MIR '07, Association for Computing Machinery, Augsburg, Bavaria, Germany, 2007, pp. 197–206. ISBN 978-1-59593-778-0. doi:10.1145/1290082.1290111.

[11] P. Harpring, Development of the Getty Vocabularies: AAT, TGN, ULAN, and CONA, *Art Documentation: Journal of the Art Libraries Society of North America* **29**(1) (2010), 67–72. doi:10.1086/adx.29.1.27949541.

[12] I. Kirton and M. Terras, Where Do Images of Art Go Once They Go Online? A Reverse Image Lookup Study to Assess the Dissemination of Digitized Cultural Heritage | MW2013: Museums and the Web 2013. https://perma.cc/CT5E-QZ77.