Article publicat / Published paper:

ABSTRACT
Evaluating image retrieval systems in a quantitative way, for example by computing measures like mean average precision, allows for objective comparisons with a ground-truth. However, in cases where ground-truth is not available, the only alternative is to collect feedback from a user. Thus, qualitative assessments become important to better understand how the system works. Visualizing the results could be, in some scenarios, the only way to evaluate the results obtained and also the only opportunity to identify that a system is failing. This necessitates developing a User Interface (UI) for a Content Based Image Retrieval (CBIR) system that allows visualization of results and improvement via capturing user relevance feedback. A well-designed UI facilitates understanding of the performance of the system, both in cases where it works well and perhaps more importantly those which highlight the need for improvement. Our open-source system implements three components to facilitate researchers to quickly develop these capabilities for their retrieval engine. We present: a web-based user interface to visualize retrieval results and collect user annotations; a server that simplifies connection with any underlying CBIR system; and a server that manages the search engine data. The software itself is described in a separate submission to the ACM MM Open Source Software Competition.

KEYWORDS
ACM proceedings, B?X, Visualization tool, image retrieval, visual searching, relevance feedback, annotation tool

ACM Reference format:

*Produces the permission block, and copyright information

1 INTRODUCTION
The evaluation of most CBIR systems (e.g. [1, 3, 4]) is based on quantitative assessment using an appropriate metric, such as mean average precision, to allow for an objective comparison between different approaches. However, qualitative assessments are often equally important, particularly from the perspective of a researcher/developing seeking to improve the underlying models and representations. Qualitative evaluation allows for exploration of success and failure cases to better understand the characteristics of the system and thus how it might be improved.

In this submission, we demonstrate an open source software framework that we have designed to facilitate researchers to qualitatively evaluate their CBIR systems. Our work implements a user interface capable of visualizing results with annotation functionality for relevance feedback. Our software is simple to use and easily adaptable to any CBIR system. It is fully open source, and so available free of change, and extensible to new functional requirements in the future. Very often developers of CBIR systems have to implement a front-end system to better understand how to improve their systems. Our framework is designed to allow them to quickly and easily integrate a front-end UI. The software itself, how to run it, and how to integrate a CBIR system, is described in detail in a separate submission to the ACM Multimedia Open Source Software Competition.

2 SYSTEM OVERVIEW
Our system is based on a web application that is able to display the similarity ranking of images retrieved for a given query and, optionally, collect user annotations in order to improve the results. The design of the system consists of a UI that is connected with the CBIR engine through a server that aims to ease the integration of any image retrieval system.

In the demonstration, examples of how to compute rankings will be shown in order to showcase all the different functionalities of the system. The CBIR system used within the framework for this purpose is an adapted version of the CBIR engine1 described in [2]. A video is also provided that illustrates typical usage of the framework 2.

There are three different ways of introducing an input image into the system as shown in figure 1: by introducing its URL, by uploading it from a file or by just selecting any of the example

1https://bitbucket.org/emoh/e-python-server-salbow/
2https://youtu.be/HubEJ43CFte
queries displayed in any dataset. The framework has three main 

modes of operation: explorer mode, annotation mode and query 

expansion mode.

- **Explorer mode**

  Explorer mode can be used to get to know datasets in a 
dynamic and interactive way. This is achieved by navigating 
through the database via "clickable" images within rankings list 
facilitating iterative querying, as shown Figure 1.

- **Query expansion vs annotation mode**

  The query expansion and annotation modes are provided 
to allow investigation of CBIR performance in two different 
ways. The most appropriate mode to use depends upon the 
average precision (AP) of a given query. Experimental results 
show that some rankings improve in accuracy by annotating 
positive images through the "query expansion" mode. Indeed, 
this approach will invariably improve performance when the 
initial AP of the query is very low, since there is a lot of scope 
for improvement.

  However, it can also be shown that the AP of a query can 
significantly improve when providing negative annotations to 
the system through the annotation mode. This is due to the 

fact that negative annotations provide information that allows 
discarding certain image classes, thus reducing the failure options 
more directly. Furthermore, this is also useful when refining a 
query, e.g. to specify one of multiple objects appearing in the 
query, as can be observed in Figure 2.

Figure 1: Figure showing the three options to input im-
ages in the system: url of an image, image from file or 
image example of a dataset.

Figure 2: Illustration of the possible use of the anno-
tation tool: in this example, the user used the annota-
tions to select one of the two objects that appeared in 
the initial query.

3 CONCLUSION

In conclusion, we provide an open source software framework that 
can be used to qualitatively evaluate and improve the performance 
of a CBIR system. Specifically, as an example of the potential bene-
fits, we show how the system we propose can significantly improve 
CBIR performance through the provision of negative annotations 
while, by using only the positive ones, it will help the system to be 
more precise without significant changes in results. Of course, this 
implies that "annotation mode" should be regulated by credentials 
or by logging in a private session in order to have a control of good 
practices when collecting feedback so as not to negatively impact 
performance. In contrast, the “query expansion” mode could be 
used to experiment freely with the system without consequence, 
as it effectively corresponds to just calculating the mean of the 
selected image descriptors.

ACKNOWLEDGMENTS

This publication has emanated from research conducted with the 
financial support of Science Foundation Ireland (SFI) under grant 
number SFI/12/RC/2289 and SFI/15/SIRG/3283, and partially sup-
ported by the Spanish Ministry of Economy and Competitivity and 
the European Regional Development Fund (ERDF) under contract 
TEC2016-75976-R.

REFERENCES

Weighting for Aggregated Deep Convolutional Features. In ECCV Workshops.
2017. Saliency Weighted Convolutional Features for Instance Search. arXiv 
O’Connor, and Xavier Giró. 2016. Bags of Local Convolutional Features for 
Scalable Instance Search. In ICMR.