Master Project at the Pattern Recognition Lab, FAU Erlangen-Nuremberg

Max Seitzer, Vincent Christlein

Report: Object retrieval system for historical documents

This project is about building an image retrieval system targeted at scans of historical documents. The background goal is to give historians a way of finding out on which documents in a corpus certain patterns appear, without strenuously having to go through them by hand. This domain is difficult for the object retrieval task for two reasons. First, the scanned documents are of high resolution, while the searched patterns are typically very small. Second, there is next to no annotated data available, making it difficult to train a feature extractor.

The approach taken in this project is relying on global features aggregated from convolutional feature maps. More specific, it is an implementation of Tolias et al. [1]. Refer to their paper for more information about the approach.

The system was implemented in Python, using Tensorflow and Keras for CNNs pretrained on ImageNet. The implementation (folder `src/`) contains of a library implementing the offline-stage feature extraction and online-stage retrieval algorithm. Additionally, there is a set of scripts (folder `cmd/`) forming the commandline interface of the library. Last, the folder `web/` contains a simple web server with a web frontend allowing to interactively query the library. Figure 2 shows a screenshot of the search interface. The readme file accompanying the code describes the setup and usage of all scripts.

In order to evaluate the system, two datasets were generated from images obtained from monasterium.net [2]. As using the original image sizes would have been to computationally expensive due to their sizes, all images were resized such that there larger side is 1000 pixels or less. The first dataset consists of 977 images of notary charters. The second, larger dataset consists of 32192 images, combining all images in the first dataset with images containing no notary charters, serving as distractor images.

To be able to judge the retrieval quality, ground truth correspondences of the same pattern appearing across different images need to be established. To this end, 606 search patterns (notary signs) were hand-annotated with bounding boxes. On average, a pattern is 129 pixels wide and 155 pixels high. Then, 295 of the patterns were sorted into 77 correspondency groups with on average 3.8 patterns per group. The remaining 311 patterns appear only once and have no corresponding patterns on other images.

Evaluation

The approach was evaluated on the two datasets using the queries as described above. For the smaller dataset, convolutional features were generated from three different networks, all pretrained on ImageNet: VGG16, VGG19, ResNet50. For the larger dataset only features from VGG16 were generated and evaluated due to computational limitations.

The retrieval performance is measured by the standard metric for this task, the mean average precision (mAP). Each single pattern of a group is used as a query, where the target of the

	VGG16	VGG19	ResNet50	VGG16 with distractors
mAP retrieval	23.56	25.74	26.53	13.68
IoU	37.32	27.33	34.25	39.77
mAP localization	8.21	5.26	6.5	5.57
Dimensionality	512	512	2048	512
Keras layer	block5_conv3	block4_conv4	activation49	block5_conv3

Figure 1: Retrieval and localization quality in percentages using convolutional features from different networks, with the layer that generated them. After retrieval, localization is performed on the best 100 images. All other parameters were used as in [1].

query consists of all images of patterns in the group (including the image the query pattern itself appears on). A target image is counted as correctly retrieved if it appears among the first N images returned by the query, where N is the amount of target patterns in the group.

For localization performance, two numbers are reported: the standard metric for localization, intersection over union (IoU), and the mAP over correctly localized patterns. For IoU, the average is reported over all patterns that were correctly retrieved. For localization mAP, a pattern is counting as correctly localized if it was correctly retrieved and has at least 50% overlap with the ground truth pattern's bounding box.

The results are shown in figure 1. Across the different models, the system achieves around 25% retrieval mAP, which means that on average, every fourth query result is a hit. This is subpar in comparison with the results the same approach achieves on datasets such as Oxford5k (77.3%) or Paris6k (86.5%) [1]. On the larger dataset with distractor images, retrieval performance drops by 10 percentage points, which is somewhat expected as the dataset size is increased by 33 times. In comparison, Tolias et al. [1] report a drop of about 5 percentage points in mAE when going to a 21 times bigger dataset. Interestingly, the used CNN model does not have a big impact on performance. Performance increases when using a more complex model such as the residual network, but only by a couple percentage points.

The retrieval performance reflects the difficulty of the domain. In standard object retrieval datasets, the object to search takes up a large fraction of the image, whereas in the domain at hand, the patterns are only covering a small part of the image. It is possible that the approach of using global image descriptors does not perform well on this domain, or needs further tuning to perform better. Another reason for the performance difference is the ImageNet dataset the CNNs were trained on: the Oxford5k and Paris6k datasets are more similar to ImageNet than the notary charters dataset, and so the computed features perform better.

For localization, the found bounding boxes cover on average around one third of the ground truth bounding box. The low localization mAP follows the bad retrieval scores: of the one fourth of images correctly retrieved, only around one forth could be correctly localized. This is probably because the main goal of the used approach is image retrieval, and localization is only a byproduct. On the larger dataset, localization performance stays constant as it is only measured on correctly retrieved images.

As the system aspires to be real-world usable, the computational performance of the approach should not be neglected. To this end, the system was benchmarked using the evaluation dataset with VGG16 features. The machine the benchmark was run on used an Intel Xeon E5-1630

CPU with 4 cores at 3.7 GHz, and a Nvidia GeForce GTX1080 GPU. On average, a query took 16.9 seconds. An analysis showed that the query time was overwhelmingly spent on localization. Implementing JIT compilation for the critical localization functions, the average query time was brought down to 9.5 seconds. Introducing parallelization across multiple cores improved the runtime to 3.46 seconds. The approach also scales well on the larger dataset, with the average query taking 4.2 seconds. This is because the main runtime is spent on localization, which is constant in the amount of images, and feature extraction, which only has to be once per query.

It should also be noted that the system in the current form requires large amounts of disk space, as the convolutional feature maps of each image in the dataset have to be stored for localization. The evaluation dataset takes approximately 6GB, which is about 6MB per image. It is expected that the space requirements can be drastically reduced by using sparse matrices and applying quantization as in [1].

Future Work and Conclusion

There are a couple of ideas of where to go from here.

Most obviously, the CNNs could be finetuned on the domain, e.g. by learning a binary classification task on pattern/no pattern. This should give a boost in performance, but unfortunately could not be implemented in this project due to time constraints. The problem with the finetuning approach is that there must be sufficient annotated data available. After this project, the notary charters dataset has some annotated patterns, but in general this is not the case for historical documents.

In the face of no labeled data, one could try features learned by unsupervised approaches. In this category fall generative adversarial networks [3], which could be trained on patches of documents. The discriminator network could then be used to provide the convolutional feature maps.

A more semi-supervised approach could be to make use of one-shot learning [4, 5]. Here, the idea is that the network only needs to see a few examples of a pattern to learn the underlying pattern statistics. Successfully applying this concept would allow the retrieval system to adapt to the queries users pose to it, by learning how queried patterns typically look like.

Another route which might prove to be successful is to learn a network for image retrieval end-to-end [6, 7]. Instead of using a network trained for a classification task, one can directly train a network to produce representations which perform well on image retrieval. This is done by training with a triplet loss, which requires to have a notion of similarity between training images. Applied to the notary charters dataset, this could be implemented by defining two images on which the same pattern appears to be similar.

Finally, an independent way to evaluate the system would be the DocExplore challenge [8], which concerns itself with pattern spotting in medieval document images. They provide a dataset together with query patterns, and baseline results which can be compared to.

In conclusion, image retrieval on historical documents is a difficult problem, and the implemented approach alone is not able to solve it sufficiently. More research needs to be done on how to learn the domain in the face of few labeled samples and how to overcome the problem that patterns are small compared to the full image. This project laid the groundwork in form of

a software framework which can be used for interactive querying and evaluation, building an evaluation dataset and providing baseline results on it.

References

[1] Giorgos Tolias, Ronan Sicre, and Hervé Jégou. Particular object retrieval with integral max-pooling of CNN activations. *CoRR*, abs/1511.05879, 2015. URL `http://arxiv.org/abs/1511.05879`.

[2] monasterium.net. `http://monasterium.net:8181/`.

[3] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative Adversarial Networks. *ArXiv e-prints*, June 2014.

[4] O. Vinyals, C. Blundell, T. Lillicrap, K. Kavukcuoglu, and D. Wierstra. Matching Networks for One Shot Learning. *ArXiv e-prints*, June 2016.

[5] D. Jimenez Rezende, S. Mohamed, I. Danihelka, K. Gregor, and D. Wierstra. One-Shot Generalization in Deep Generative Models. *ArXiv e-prints*, March 2016.

[6] A. Gordo, J. Almazan, J. Revaud, and D. Larlus. End-to-end Learning of Deep Visual Representations for Image Retrieval. *ArXiv e-prints*, October 2016.

[7] A. Gordo, J. Almazan, J. Revaud, and D. Larlus. Deep Image Retrieval: Learning global representations for image search. *ArXiv e-prints*, April 2016.

[8] S. En, S. Nicolas, C. Petitjean, F. Jurie, and L. Heutte. New public dataset for spotting patterns in medieval document images. *Journal of Electronic Imaging*, 26(1):011010, January 2017. doi: 10.1117/1.JEI.26.1.011010. URL `http://spotting.univ-rouen.fr/`.

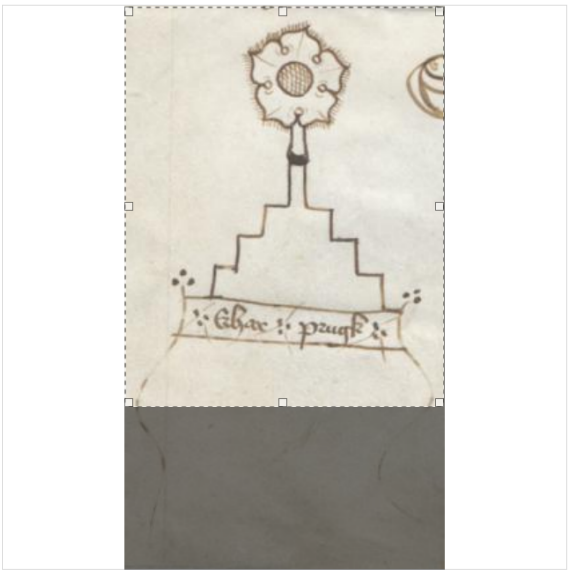
Historical Image Retrieval

Choose an image

Upload or enter image URL

Get image

Choose a selection



WxH: 240x422 x: 0:240, y: 0:300

Current selection:

Number of results: 50

Localization Rerank

Average Query Expansion

Search with selection

Search results

Retrieved 50 results in 5.825 seconds.

- Name: BayHStA-HUPassau_14421221_01767_r.jpg
- Score: 0.6193
- Box: 65:290, 470:778
- External link

- Name: BayHStA-HUPassau_14420821_01760_r.jpg
- Score: 0.5911
- Box: 48:563, 323:903
- External link

- Name: BayHStA-HUPassau_14510730_01905_r.jpg
- Score: 0.5465
- Box: 65:661, 32:776
- External link

- Name: BayHStA-HUPassau_14421009_01762_r.jpg
- Score: 0.4809
- Box: 161:613, 0:556
- External link

Figure 2: The web frontend with query selection and result section