

Discovering Visual Patterns in Art Collections with Spatially-consistent Feature Learning

Xi Shen, Alexei A. Efros, Mathieu Aubry

▶ **To cite this version:**

Xi Shen, Alexei A. Efros, Mathieu Aubry. Discovering Visual Patterns in Art Collections with Spatially-consistent Feature Learning. 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Jun 2019, Long Beach, CA, United States. pp.9270-9279, 10.1109/CVPR.2019.00950 . hal-02104041

HAL Id: hal-02104041

https://enpc.hal.science/hal-02104041

Submitted on 16 Feb 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

This CVPR paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Discovering Visual Patterns in Art Collections with Spatially-consistent Feature Learning

Xi Shen[1], Alexei A. Efros[2], and Mathieu Aubry[3]

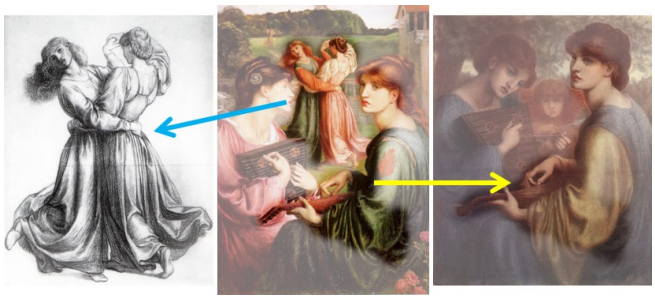
[1,3]LIGM (UMR 8049), École des Ponts ParisTech
[2]University of California, Berkeley
[1,3]{xi.shen, mathieu.aubry}@enpc.fr, [2]efros@eecs.berkeley.edu

Abstract

Our goal in this paper is to discover near duplicate patterns in large collections of artworks. This is harder than standard instance mining due to differences in the artistic media (oil, pastel, drawing, etc), and imperfections inherent in the copying process. Our key technical insight is to adapt a standard deep feature to this task by fine-tuning it on the specific art collection using self-supervised learning. More specifically, spatial consistency between neighbouring feature matches is used as supervisory fine-tuning signal. The adapted feature leads to more accurate style-invariant matching, and can be used with a standard discovery approach, based on geometric verification, to identify duplicate patterns in the dataset. The approach is evaluated on several different datasets and shows surprisingly good qualitative discovery results. For quantitative evaluation of the method, we annotated 273 near duplicate details in a dataset of 1587 artworks attributed to Jan Brueghel and his workshop[1]. Beyond artworks, we also demonstrate improvement on localization on the Oxford5K photo dataset as well as on historical photograph localization on the Large Time Lags Location (LTLL) dataset.



(a) common detail (dog) discovered in our new Brueghel dataset of 1587 artworks



(b) relationship between painting and two studies discovered from collection of 195 artworks by Dante Gabriel Rossetti

Figure 1: Examples of repeated visual patterns automatically discovered by our algorithm. Sources: (a) left: *Nymphs Sleeping After the Hunt, Spied on by Satyr* (oil), center: *Diana's Nymphs After the Hunt* (oil), right: *Seventeen Studies of Different Dogs* (drawing); (b) *The Bower Meadow* (left: chalk, center: oil, right: pastel)

1. Introduction

Visiting a world-class art museum might leave one with an impression that each painting is absolutely unique and unlike any other. In reality, things are more complicated. While working on a painting, an artist would typically create a number of preliminary sketches and studies to experiment with various aspects of the composition. Many of these studies also find their way into (usually more provincial) museums. Some artists enjoy returning time and again to the same subject matter (e.g. Claude Monet and his series of 25 paintings of the same haystacks). Moreover, during the Renaissance, it was not uncommon for an artist (or an apprentice in his workshop) to reuse the same visual elements or motifs (an angel, a cart, a windmill, etc) in multiple paintings, with little or no variation. For example, Flemish painter Jan Brueghel is believed

[1]Code and data is available at http://imagine.enpc.fr/~shenx/ArtMiner

to have created many paintings that were imitations, pastiches, or reworkings of his own works, as well as these of his father, Pieter Breughel the Elder [25]. Art historians are keenly interested in mapping out such visual connections between artworks to discover influences, find provenance, and even establish authorship. Currently, this is being done entirely by hand, with researchers spending months or even years in museum archives hoping to discover common visual patterns.

This paper presents an approach for automatically discovering repeated visual patterns in art collections, as shown on Figure 1. We propose to learn a deep visual feature able to find correspondences between near-duplicate visual elements across different artworks. This task is quite challenging, requiring a feature that is both highly discriminative (i.e. tuned to find copies of the same object instance rather then samples of an object category), but also invariant to changes in color, style, artistic media, geometric deformation, etc. Manually collecting and labelling a large enough artwork dataset containing enough variability would require enormous effort by professional art historians, which is exactly what we are trying to avoid. Therefore, we propose a method which learns in a self-supervised way, adapting a deep feature to a given art collection without any human labelling. This is done by leveraging neighbourhood spatial consistency across matches as free supervisory signal.

Using our trained feature, we demonstrate that a simple voting and regression procedure, in line with classic verification step of instance recognition [33], lets us discover visual patterns that are repeated across artworks within the dataset.

We demonstrate our visual pattern discovery approach on several collections of artwork, including a new annotated dataset of 1587 works attributed to the Brueghel family. To further evaluate the generality of our method, we have also used it to perform localization by matching a set of reference photographs to a test image. We demonstrate that our feature learning procedure provides a clear improvement on the Oxford5K dataset [33], and results in state-of-the-art performance on the Large Time Gap Location dataset [16] for localizing historical architecture photographs with respect to modern ones.

Our main contributions are: 1) a self-supervised approach to learn a feature for matching artistic visual content across wide range of styles, media, etc; 2) the introduction of a large new dataset for evaluating visual correspondence matching; 3) an approach to discover automatically repeated elements in artwork collections.

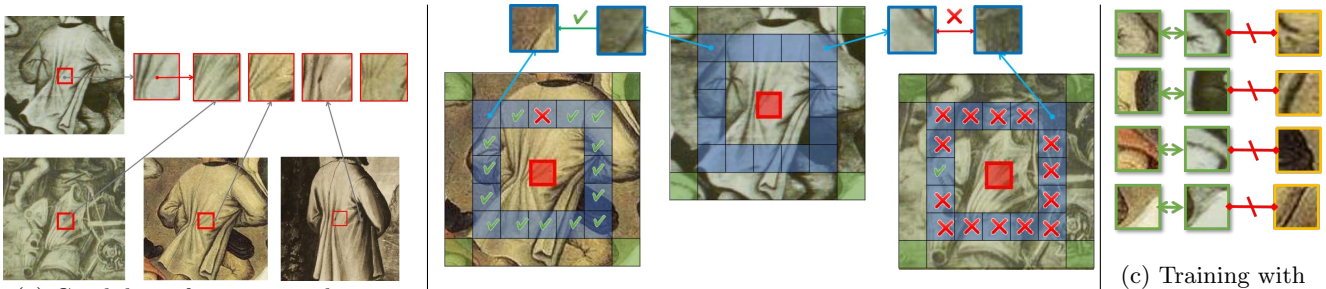
2. Related Work

Computer vision and art. There is a long standing and fruitful collaboration between computer vision and art. On the synthesis side, promising results have been obtained for transferring artistic style to a photograph [24, 17, 50], or even trying to create art [14, 23]. On the analysis side, there are several efforts on collection and annotation of large-scale art datasets [27, 31, 34, 48, 44, 30], and using them for genre and authorship classification [27, 45, 44]. Others focus on applying and generalizing visual correspondence and object detection methods to paintings using both classical [41, 18, 7, 4], as well as deep [6, 8, 47, 20] methods. Most closely related to us is work of Yin et al [49], which used the same Brueghel data [1], annotating it to train detectors for five object categories (carts, cows, windmills, rowboats and sailbaots).

Our goal, however, is to go further and focus on the computational analysis of relationships *between* individual artworks. Seguin et al [40, 39] propose to find visual relationships in collections of paintings. However, while they use off-the-shelf CNNs trained in a supervised manner, we focus on the design of a new self-supervised feature learning specifically trained for the task. This allows us to focus on near-exact reproduction of detail, rather than a more generic visual similarity, which is what most art historians are actually interested in regards to specific corpora, such as the works of Brueghel family [25].

Spatial consistency as supervisory signal. Spatial consistency is a widely used signal in many computer vision tasks from geometry to retrieval. The classic work of Sivic et al. [43] performs instance retrieval based on the extraction of spatially consistent local feature matches. This direction has been further developed with specially adapted features for place recognition across large visual changes [16, 21, 46, 4]. Beyond instances, this idea has been extended to discovering object categories [5] and their segmentations [38]. Our discovery of repeated patterns through correspondence consistency is reminiscent of the line of work on mid-level visual element discovery [42, 9, 10]. These idea have been used in the context of temporal and spatial image collection analysis, to discover the elements characteristic of a specific location [12], or the evolution of these elements over time [26].

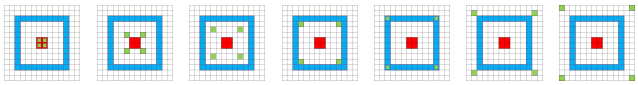
Spatial consistency has also been used to learn deep visual features for object category in a self-supervised way, either by predicting the spatial configuration of patches [11] or predicting the patch given its context [32]. In a similar spirit, Rocco et al. [37] recently demonstrated how to learn visual representa-



(a) Candidates from proposal region (b) Selection with verification regions (c) Training with positive regions

Figure 2: Feature Learning Strategy. (a) Our approach relies on candidate correspondences obtained by matching the features of a proposal region (in red) to the full database . (b) The candidate correspondences are then verified by matching the features of the verification region (in blue) of the query in the candidate images and checking for consistency . (c) Finally, we extract features from the positive regions (in green) from the verified candidates and use them to improve the features using a metric learning loss.

tions through geometric consistency to predict object-category-level correspondences between images. We, on the other hand, aim at learning features for matching only stylistically different versions of the same instance.



P2 P4 P6 P8 P10 P12 P14

Figure 3: Different region configurations. (red: proposal regions, blue: verification regions, green: positive regions)

3. Dataset-specific Feature Learning

This section describes our strategy for adapting deep features to the task of matching artworks across styles in a specific dataset. Starting with standard ImageNet pre-trained deep features, our idea is to extract hard-positive matching regions form the dataset that we then use in a metric learning approach to improve the features. Our two key hypothesis are that: (i) our dataset includes large parts of images that are copied from each other but are depicted with different styles, and (ii) the initial feature descriptor is good enough to extract some positive matches. Our training thus alternates between two steps that we describe bellow: (1) mining for hard-positive training samples in the dataset based on the current features using spatial consistency, and (2) updating the features by performing a single gradient step on the selected samples.

3.1. Mining for Positive Feature Pairs

For our approach to work, it is crucial to select positive matching examples that are both accurate and difficult. Indeed, if the features are trained with false matches, training will quickly diverge, and if the matches are too easy, no progress will be made.

To find these hard-positive matching features, we rely on the procedure visualized in Figure 2.

Candidate sampling. *Proposal regions* are randomly sampled from each image in the dataset to be used as query features. These are matched densely at every scale to all the images in the dataset using cosine similarity in feature space. This can be done efficiently and in parallel for many queries using a normalization and a convolution layer, with the weights of the convolution defined by the query features. For each query we select one of its top K matches as candidate correspondences (Figure 2a). These candidates contain a high proportion of bad matches, since most of the queries are likely not repeated K times in the dataset and since our feature is imperfect.

Candidate verification. To verify the quality of candidate matches given by the previous step, we rely on special consistency: a match will be considered valid if its neighbours agree with it. More precisely, let's assume we have a candidate match between features from the proposal region p_A in image A and a corresponding region p_B in image B, visualized in red in Figure 2b and 3. We define a *verification region* around p_A, visualized in blue. Every feature in this region is individually matched in image B, and votes for the candidate match if it matched consistently with p_B. Summing the votes of all the features in the verification region allows us to rank the candidate matches. A fixed percentage of the candidates are then considered verified. The choice of the verification region is, of course, important to the success of this verification step. The key aspect is that the features in the verification region should be, as much as possible, independent of the features in the proposal region. On the other hand, having them too far apart would reduce the chances of the region being completely matched. For our experiments, we used the 10x10 feature square centred around the query region (Figure 3).

Generating hard positives. Finally, given a set of verified correspondences, we have to decide which features to use as positive training pairs. One possibility would be to directly use features in the proposal region, since they have been verified. However, since the proposal region has already been "used" once (to verify the matches), it does not bring enough independent signal to make quality hard positives. Instead, we propose to sample positives from a different *positive region*. We evaluated different configurations for the positive region, as visualized in Figure 3 (in green). We choose to keep only 4 positive pairs per verified proposal, positioned at the corners of a square and denote the different setups as P2 to P14, the number corresponding to the size of the square. We found that P12 and P14 perform better than the alternatives.

3.2. Feature Fine-tuning

After each selection of positive feature pairs (Figure 2b in green), we update of our feature (Figure 2c) using a single gradient step of the following triplet metric learning loss:

$$L(P_1, P_2, \{N_i\}) = -\min(\lambda, s(P_1, P_2)) + \frac{1}{N_{neg}} \sum_{i=1}^{N_{neg}} \max(s(P_1, N_i), 1-\lambda) \quad (1)$$

where P_1 and P_2 are corresponding features in the positive regions, $\{N_i\}_{\{i=1,2...N_{neg}\}}$ are negative samples, s is the cosine similarity metric and λ is a hyperparameter. We select the negatives as the set of top matches to P_1 in P_2's image. This selects hard negatives and avoids any difference in the distribution of the depiction styles in our positive and negative samples. We chose a relatively high number of negatives N_{neg} to account for the fact that some of them might in fact correspond to matching regions, for example in the case of repeated elements, or for location near the optimal match.

Implementation details. In all of our experiments, we used *conv4* features of the ResNet-18 [22] architecture. We resized all images such that their maximum spatial dimension in the feature map was 40, leading to approximately 1k features per image at the maximum scale. For each image, we used 7 different scales, regularly sampled on two octaves with 3 scales per octave. For positive sampling, we used square queries of 2×2 features, $K = 10$ candidate matches for each query. From these candidate, the top 10% with the most votes from neighbours were considered verified. Note that these parameters might need to be adjusted depending on the diversity and size of the dataset, but we found that they performed well both for the Brueghel [1] and

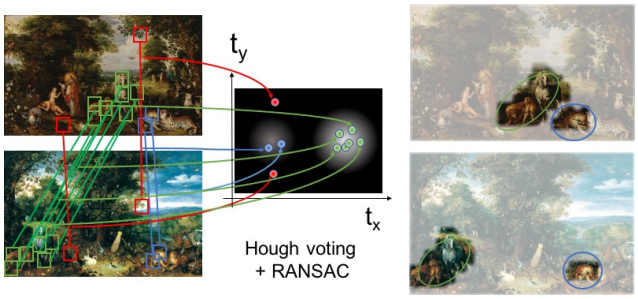


Figure 4: Discovery through geometric consistency.

LTLL [16] datasets. For training on Oxford5K [33] dataset, the query patches are only sampled inside the annotaed bounding boxes of the 55 query images, and we only find candidate matches in 2000 images randomly sampled in the whole dataset. The hyperparameters of the triplet loss, N_{neg} and λ, are fixed to 20 and 0.8 respectively. Our models were trained with the Adam [28] optimizer with learning rate 1e-5 and $\beta = [0.9, 0.99]$. Using a single GPU Geforce GTX 1080 Ti, training converged in approximately 10 hours, corresponding to 200 iterations of the feature selection and training. Most of the time is spent extracting and verifying candidate matches. ImageNet pre-training was used for initialization in all experiments.

4. Spatially consistent pattern mining

In this section, we describe how our algorithm discovers repeated patterns in a dataset given style-invariant features learned in the previous section. We follow the classic geometric verification approach [33]: for all pairs of images in the dataset, we first compute features and match them between the two images, then select consistent matches, and finally find image regions that have many consistent feature matches. This allows us to build a graph between all corresponding image regions, from which we extract clusters of related images and repeated elements. In the following, we present briefly each of these steps.

4.1. Identifying region correspondences

Our discovery procedure for a pair of images is visualized in Figure 4. We start by computing dense correspondences between the two images using our learned feature. These will be quite noisy. We first use Hough voting to identify potential groups of consistent correspondences. As a first approximation, each correspondence votes for a translation and change in scale. We then extract the top 10 translation candidates, and, using a permissive threshold, focus on the correspondences in each group independently. Within each group, we use RANSAC to recover an affine transformation and the associated inliers. This allows to ac-

count for some deformation in the copy process, but also variations in the camera viewpoint with respect to the artwork.

4.2. Scoring correspondences

After deformations between image regions are identified, we score the correspondence based both on the quality of the match between the features and geometric criteria. We use the following weighted average of the feature similarity:

$$S(\mathcal{I}) = \frac{1}{N} \sum_{i \in \mathcal{I}} e^{(-\frac{e_i^2}{2\sigma^2})} s_i \tag{2}$$

where $\mathcal{I}$ is the index of the inlier correspondences, e_i is the error between correspondence i and the geometric model, s_i the similarity of the associated descriptors and $\frac{1}{N}$ is normalization by the number of features in the source image.

4.3. Correspondence graph

Using the score S, we can separate our dataset into clusters of connected images. These clusters are already interesting and visually appealing, especially for dataset with few repeated details. However, to avoid obtaining very large clusters when many details are repeated in overlapping sets of images, one needs to individually identify each detail region. To do that we built a graph from all the connected image pairs. The nodes are the regions that are mapped between pairs of images. Each image can contain several overlapping regions. We connect regions that are matched to each other as well as regions in the same image that overlap with an Intersection over Union (IoU) score greater than a given threshold (0.5 in our experiments). Finally, we extract the connected components in this graph. Each of them corresponds to a different detail that is repeated in all images of the group.

5. Experiments

In this section, we analyse and evaluate our approach. We first present the main datasets we used, including our new annotations of the Brueghel dataset [1] specifically targeted toward the new task we propose. Second, we provide detailed results and analysis for the task of one-shot visual pattern detection. Finally, we present quantitative and qualitative results for our discovery procedure.

5.1. Datasets

Brueghel. We introduce new annotations for the Brueghel dataset [1], which are available on our project webpage. Indeed, to the best of our knowledge, no other annotation for the task of near duplicate detection in artwork is currently available.

The Brueghel dataset contains 1,587 artworks done in different media (e.g. oil, ink, chalk, watercolour) and on different materials (e.g. paper, panel, copper), describing a wide variety of scenes (e.g. landscape, religious, still life) This dataset is especially adapted for our task since it assembles paintings from artists related to the same workshop, who thus had many interaction with each other, and includes many copies, preparatory drawings, and borrowed details. With the help of our art history collaborators, we selected 10 of the most commonly repeated details in the dataset and annotated the visual patterns in the full dataset using the VGG Image Annotator tool [13]. Exemples of the 10 annotated patterns can be seen in Figure 5 as queries (blue boxes). We were careful to select diverse patterns, and for each of them to annotate only duplicates, and not full object classes. Note for example that for the horses and lion classes, we annotated separately two variants of the details (front and back facing lion, front and right facing horse). This resulted in 273 annotated instances, with a minimum of 11 and a maximum of 57 annotations per pattern.

These annotations allow us to evaluate one-shot duplicate detection results. In our evaluation, we use an IoU threshold of 0.3 for positives, because precise annotations of the bounding boxes in different environment is difficult and approximate detections would be sufficient for most applications. In practice, our detected bounding boxes, visualised in Figure 5 (green boxes) often appear more consistent than the annotations. We compute the Average Precision for each query, average them per class and report class level mean Average Precision (mAP).

Large Time Lags Locations (LTLL). While our discovery algorithm targets copy detection in art, it should can detect the same object instances in photographs. We test our algorithm on the LTLL [16] dataset. It contains historic and modern photographs captured from 25 locations over a time interval of more than 150 years. In total the dataset contains 225 historical and 275 modern photographs of the same locations. The task proposed in LTLL is to recognize the location of an old picture using annotated modern photographs. We use our discovery procedure to find the images most similar to the query. As in the original paper [16], we report classification accuracy.

Oxford5K. We also evaluate our approach on Oxford buildings [33] dataset. The dataset contains 5062 images for 11 different landmarks. We follow the stan-

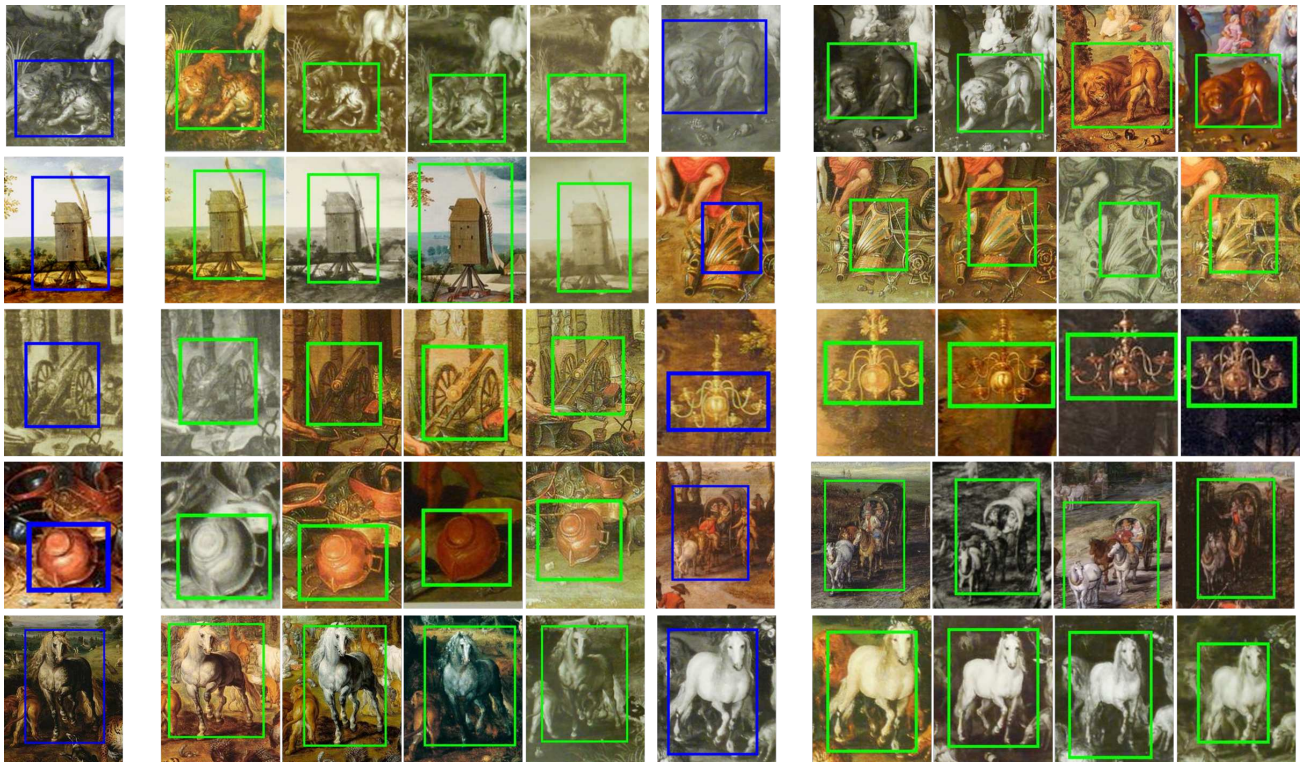


Figure 5: Detection example with our trained features on the Brueghel dataset. We show the top 4 matches (in green) for one example of query from each of our 10 annotated categories. Notice how the matches style can be different from the one of the query.



Figure 6: From a single query, shown on the left, we show the detection results obtained with cosine similarity with ImageNet feature (a) and our trained features (b) as well as the ones (c) obtained with our features and the discovery score presented in section 4.2.

dard evaluation protocol and report retrieval mAP for the 55 queries.

DocExplore. The DocExplore dataset [15], which is dedicated to spotting repeated patterns in manuscripts, is the closest existing dataset related to our task and provides extensive comparisons. However the repeated patterns in this dataset are rare and small, all exactly in the same style, with the same colors, and most of the data is text. We thus used it to validate our baseline one-shot detection approach, but could not use it for feature training. DocExplore contains over 1500 images with 1464 instances of 35 different details. For our experiments, we only considered the 18 largest details (the other ones corresponding to small letters).

WikiArt. To show the generality of our approach, we ran our discovery algorithm on paintings of other artists (Peter Paul Rubens, Dante Gabriel Rossetti and Canaletto) that we collected from WikiArt [2, 3] (respectively 387, 195 and 166 artworks).

5.2. One-shot detection

We evaluated our feature learning strategy using one-shot detection. This was performed simply by computing densely features on the dataset and computing their cosine similarity with the features corresponding to the query. The query was resized so its largest dimension in the feature map would be 8. Note that unlike standard deep detection approaches [19, 36], we do not use region proposals because we want to be able to match regions which do not correspond to objects.

Examples results using this approach for each of the 10 details we annotated on the Brueghel dataset are shown in Figure 5. It gives a sense of the difficulty of the task we target and the quality of the results we obtain. Note for example how the matches are of different styles, and how the two types of lions (top row) and the two types of horses (bottom row) are differen-

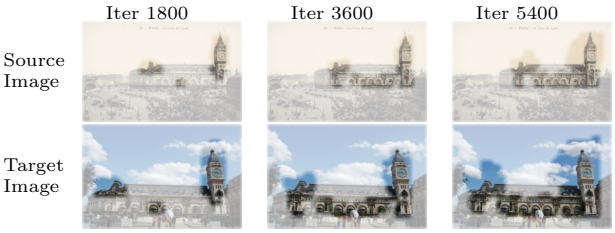


Figure 7: Discovery between images during training.

tiated. In the following, we compare these results with baselines and analyse the differences.

Validation on DocExplore. To validate that our one-shot detection approach is competitive with classical methods for finding repeated details, we ran it on the DocExplore dataset with ResNet-18 features trained on ImageNet. Our cosine-similarity based dense approach resulted in mAP of 55% on the 18 categories we considered, a clear improvement compared to the best performance of 40% obtained in [15] with classical approaches.

Feature \ Method	Cosine similarity	Discovery score
ImageNet pre-taining	58.0	54.8
Context Prediction [11]	58.8	64.29
Ours (trained on Brueghel)	**75.3**	**76.4**
Ours (trained on LTLL)	65.2	69.95

Table 1: Experimental results on Brueghel, IoU > 0.3.

Comparison and analysis on Brueghel. Here, we compare the one shot detection performance with different features using cosine similarity and the score described in equation 2. In Figure 6, we present the top 4 matches from the same query using different approaches. On this example, it can be seen that while ImageNet feature only gets the matches in similar styles, our trained feature obtains duplicated horses in different styles, showing that the learned feature is more invariant to style. Moreover, the matching can still be improved with the discovery score. The corresponding quantitative results are presented in Table 1 and confirm these observations. Indeed learning features improves the score by 17%. The discovery procedure and score provides a small additional boost, which is a first validation of our discovery procedure. We also report results for two baselines that we re-implemented: the classical Video Google [43] approach and deep feature learnt with Context Prediction [11]. With the Video Google [43] baseline, we obtained only 21.53% as retrieval mAP, showing the difficulty to address our task with SIFT [29] features. For Context Prediction, we trained the network using the Brueghel dataset and the same ResNet18 architecture and ImageNet initialization as for our method. We only obtain an improvement of 0.8% compared to ImageNet feature, much lower than the 17% provided by our method. Interestingly, training our feature on the LTLL dataset also gave a boost in performance compared to the ImageNet feature, but is clearly worst than training on the Brueghel data, showing the dataset specific nature of our training.

Method	LTLL (%)	Oxford (%)
B. Fernando et al.[16]	56.1	-
F. Radenović et al.[35]	-	87.8
ResNet18 max-pool, image level	59.8	14.0
ResNet18 + discovery	80.9	85.0
Ours (trained LTLL + discovery)	**88.5**	83.6
Ours (trained Oxford + discovery)	85.6	**85.7**

Table 2: Classification accuracy on LTLL and retrieval mAP on Oxford5K

5.3. Visual Pattern Discovery

In this part, we focus more specifically on our discovery procedure, show qualitative results on various datasets, and evaluate quantitatively for place recognition on the LTLL dataset.

Training visualization. To visualise the influence of the feature training for our discovery task, we selected a pairs of matching images and ran discovery on them with the features at different steps of training. Figure 7 visualises the results on a pair of images from the LTLL dataset. During training, larger and larger parts of the images can be matched in a consistent way, and be discovered as similar elements by our method. This shows both the efficiency of our feature training and its relevance for our task.

Quantitative analysis on one-shot localization. We evaluate our approach on one-shot localization for both the LTLL and Oxford5K datasets. The results are reported in Table 2. We compare our discovery score to cosine similarity with standard max-pooled features as well as the state of the art results of [16] on LTLL and [35] on Oxford5K.

On LTLL, we use the class of the nearest neighbour in modern photographs to localise the historical images. Using the discovery score provides a very important boost compared to the results of [16] and the max-pooled features. Using our fine-tuning procedure on the LTLL dataset improves again the results, demonstrating again the interest of our proposed dataset specific fine-tunning procedure.

Similarity, on the Oxford5K dataset, we obtain an important boost using the discovery score compared to cosine similarity with max-pooled features. Fine-tuning the features on Oxford5K improves the mAP by 0.7%. This improvement is less important than on LTLL, which is expected since there is no specific do-

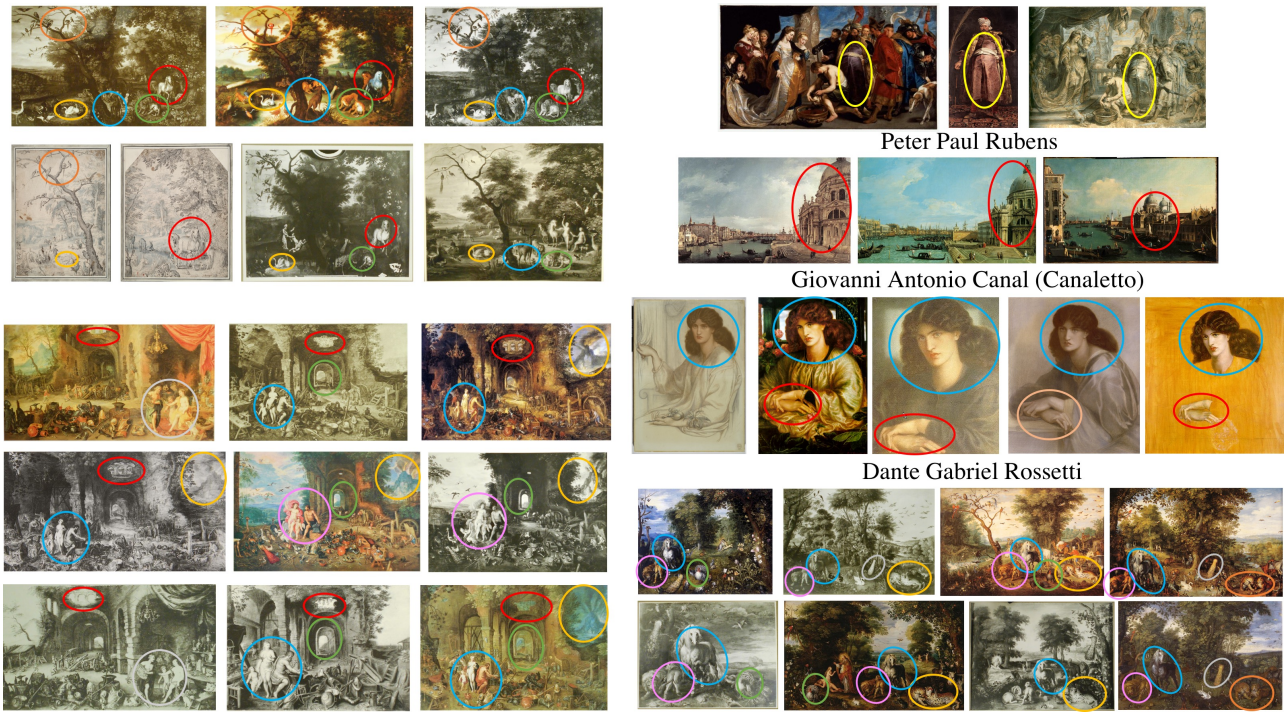


Figure 8: Example of image clusters discovered by our method. The three cluster without artist names correspond to data from the Brueghel dataset [1]

main gap between queries and targets in the Oxford5K dataset. Our result on Oxford5K is also comparable to the state-of-the-art result obtained in [35] which performs fine-tuning on a large image collection with ResNet101. As expected the retrieval mAP is better when fine-tuning on the Oxford dataset than on LTLL.

Qualitative results. We show example of our discovery results in Figure 8. More results are available in our project webpage.



Figure 9: Failure examples in paintings of the Brughel dataset and from Peter Paul Rubens.

6. Limitations and Discussion

When performing discovery on different datasets, we observed some interesting failure modes visualized in Figure 9. In the Brughel dataset, we discovered the identical circular frame of a set of paintings as a repeated pattern, as well as matched the faces in a set of similar but not identical paintings from portrait collections of Peter Paul Rubens.

More generally our method has several limitations. First, the time to perform discovery is important, which limits the size of the datasets we can handle to a few thousands images. Second, our discovery procedure relies on an affine transformation model, which might not always be rich enough. Finally, our feature learning requires having access to a dataset which includes many repeated patterns and a good feature initialization.

7. Conclusion

We have introduced a new approach to adapt features for instance matching on a specific dataset without human supervision. We have demonstrated quantitatively the promise of our method both for one-shot cross-modal detection and for cross-modal instance discovery. Last but not least, we demonstrate diverse near duplicate discovery results in several artwork datasets.

Acknowledgments This work was partly supported by ANR project EnHerit ANR-17-CE23-0008, project Rapid Tabasco, NSF IIS-1633310, Berkeley-France funding, and gifts from Adobe to Ecole des Ponts. We thank Shiry Ginosar for her advice and assistance, and Xiaolong Wang, Shell Hu, Minsu Cho, Pascal Monasse and Renaud Marlet for fruitful discussions, and Kenny Oh, Davienne Shields and Elizabeth Alice Honig for thier help on defining the task and building the Brueghel dataset.

References

[1] Brueghel family: Jan brueghel the elder." the brueghel family database. university of california, berkeley. http://www.janbrueghel.net/. Accessed: 2018-10-16. 2, 4, 5, 8

[2] Wikiart. https://www.wikiart.org/. Accessed: 2018-10-16. 6

[3] Wikiart retriever. https://github.com/lucasdavid/wikiart/. Accessed: 2018-10-16. 6

[4] M. Aubry, B. C. Russell, and J. Sivic. Painting-to-3d model alignment via discriminative visual elements. *ACM Transactions on Graphics (ToG)*, 33(2):14, 2014. 2

[5] M. Cho, S. Kwak, C. Schmid, and J. Ponce. Unsupervised object discovery and localization in the wild: Part-based matching with bottom-up region proposals. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1201–1210, 2015. 2

[6] E. J. Crowley, O. M. Parkhi, and A. Zisserman. Face painting: querying art with photos. In *BMVC*, pages 65–1, 2015. 2

[7] E. J. Crowley and A. Zisserman. Of gods and goats: Weakly supervised learning of figurative art. In *British Machine Vision Conference*, 2013. 2

[8] E. J. Crowley and A. Zisserman. The art of detection. In *European Conference on Computer Vision*, pages 721–737. Springer, 2016. 2

[9] C. Doersch, A. Gupta, and A. A. Efros. Mid-level visual element discovery as discriminative mode seeking. In *Advances in neural information processing systems*, pages 494–502, 2013. 2

[10] C. Doersch, A. Gupta, and A. A. Efros. Context as supervisory signal: Discovering objects with predictable context. In *European Conference on Computer Vision*, 2014. 2

[11] C. Doersch, A. Gupta, and A. A. Efros. Unsupervised visual representation learning by context prediction. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1422–1430, 2015. 2, 7

[12] C. Doersch, S. Singh, A. Gupta, J. Sivic, and A. Efros. What makes paris look like paris? *ACM Transactions on Graphics*, 31(4), 2012. 2

[13] A. Dutta, A. Gupta, and A. Zissermann. VGG image annotator (VIA). 5

[14] A. Elgammal, B. Liu, M. Elhoseiny, and M. Mazzone. Can: Creative adversarial networks, generating" art" by learning about styles and deviating from style norms. *arXiv preprint arXiv:1706.07068*, 2017. 2

[15] 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, 2016. 6, 7

[16] B. Fernando, T. Tommasi, and T. Tuytelaars. Location recognition over large time lags. *Computer Vision and Image Understanding*, 139:21–28, 2015. 2, 4, 5, 7

[17] L. A. Gatys, A. S. Ecker, and M. Bethge. Image style transfer using convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2414–2423, 2016. 2

[18] S. Ginosar, D. Haas, T. Brown, and J. Malik. Detecting people in cubist art. In *Workshop at the European Conference on Computer Vision*, pages 101–116. Springer, 2014. 2

[19] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014. 6

[20] N. Gonthier, Y. Gousseau, S. Ladjal, and O. Bonfait. Weakly supervised object detection in artworks. *arXiv preprint arXiv:1810.02569*, 2018. 2

[21] D. C. Hauagge and N. Snavely. Image matching using local symmetry features. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 206–213. IEEE, 2012. 2

[22] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 4

[23] A. Hertzmann. Can computers create art? In *Arts*, volume 7, page 18. Multidisciplinary Digital Publishing Institute, 2018. 2

[24] A. Hertzmann, C. E. Jacobs, N. Oliver, B. Curless, and D. H. Salesin. Image analogies. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 327–340. ACM, 2001. 2

[25] E. Honig. *Jan Brueghel and the Senses of Scale*. Pennsylvania State University Press, 2016. 2

[26] Y. Jae Lee, A. A. Efros, and M. Hebert. Style-aware mid-level representation for discovering visual connections in space and time. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1857–1864, 2013. 2

[27] S. Karayev, M. Trentacoste, H. Han, A. Agarwala, T. Darrell, A. Hertzmann, and H. Winnemoeller. Recognizing image style. *arXiv preprint arXiv:1311.3715*, 2013. 2

[28] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 4

[29] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004. 7

[30] H. Mao, M. Cheung, and J. She. Deepart: Learning joint representations of visual arts. In *Proceedings of the 2017 ACM on Multimedia Conference*, pages 1183–1191. ACM, 2017. 2

[31] T. Mensink and J. Van Gemert. The rijksmuseum challenge: Museum-centered visual recognition. In *Proceedings of International Conference on Multimedia Retrieval*, page 451. ACM, 2014. 2

[32] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros. Context encoders: Feature learning by inpainting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2536–2544, 2016. 2

[33] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Object retrieval with large vocabularies and fast spatial matching. In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pages 1–8. IEEE, 2007. 2, 4, 5

[34] D. Picard, P.-H. Gosselin, and M.-C. Gaspard. Challenges in content-based image indexing of cultural heritage collections. *IEEE Signal Processing Magazine*, 32(4):95–102, 2015. 2

[35] F. Radenović, G. Tolias, and O. Chum. Fine-tuning cnn image retrieval with no human annotation. *arXiv preprint arXiv:1711.02512*, 2017. 7, 8

[36] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015. 6

[37] I. Rocco, M. Cimpoi, R. Arandjelović, A. Torii, T. Pajdla, and J. Sivic. Neighbourhood consensus networks. *arXiv preprint arXiv:1810.10510*, 2018. 2

[38] M. Rubinstein, A. Joulin, J. Kopf, and C. Liu. Unsupervised joint object discovery and segmentation in internet images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1939–1946, 2013. 2

[39] B. Seguin, I. diLenardo, and F. Kaplan. Tracking transmission of details in paintings. In *DH*, 2017. 2

[40] B. Seguin, C. Striolo, F. Kaplan, et al. Visual link retrieval in a database of paintings. In *European Conference on Computer Vision*, pages 753–767. Springer, 2016. 2

[41] A. Shrivastava, T. Malisiewicz, A. Gupta, and A. A. Efros. Data-driven visual similarity for cross-domain image matching. *ACM Transaction of Graphics (TOG) (Proceedings of ACM SIGGRAPH ASIA)*, 30(6), 2011. 2

[42] S. Singh, A. Gupta, and A. A. Efros. Unsupervised discovery of mid-level discriminative patches. In *Computer Vision–ECCV 2012*, pages 73–86. Springer, 2012. 2

[43] J. Sivic and A. Zisserman. Video google: A text retrieval approach to object matching in videos. In *null*, page 1470. IEEE, 2003. 2, 7

[44] G. Strezoski and M. Worring. Omniart: multi-task deep learning for artistic data analysis. *arXiv preprint arXiv:1708.00684*, 2017. 2

[45] 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 *Image Processing (ICIP), 2016 IEEE International Conference on*, pages 3703–3707. IEEE, 2016. 2

[46] Y. Verdie, K. Yi, P. Fua, and V. Lepetit. Tilde: A temporally invariant learned detector. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5279–5288, 2015. 2

[47] N. Westlake, H. Cai, and P. Hall. Detecting people in artwork with cnns. In *European Conference on Computer Vision*, pages 825–841. Springer, 2016. 2

[48] M. J. Wilber, C. Fang, H. Jin, A. Hertzmann, J. Collomosse, and S. J. Belongie. Bam! the behance artistic media dataset for recognition beyond photography. In *ICCV*, pages 1211–1220, 2017. 2

[49] R. Yin, E. Monson, E. Honig, I. Daubechies, and M. Maggioni. Object recognition in art drawings: Transfer of a neural network. In *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*, pages 2299–2303. IEEE, 2016. 2

[50] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. *arXiv preprint*, 2017. 2